

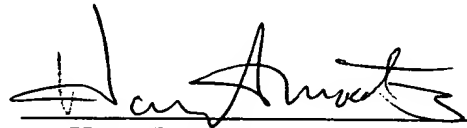
**BEFORE THE OFFICE OF ENROLLMENT AND DISCIPLINE
UNITED STATES PATENT AND TRADEMARK OFFICE**

LIMITED RECOGNITION UNDER 37 CFR § 10.9(b)

Mr. Haruo Yawata is hereby given limited recognition under 37 CFR § 10.9(b) as an employee of Beyer Weaver & Thomas, LLP to prepare and prosecute patent applications wherein the patent applicant is the client of Beyer Weaver & Thomas, LLP, and the attorney or agent of record in the applications is a registered practitioner who is a member of Beyer Weaver & Thomas, LLP. This limited recognition shall expire on the date appearing below, or when whichever of the following events first occurs prior to the date appearing below: (i) Mr. Haruo Yawata ceases to lawfully reside in the United States, (ii) Mr. Haruo Yawata's employment with Beyer Weaver & Thomas, LLP ceases or is terminated, or (iii) Mr. Haruo Yawata ceases to remain or reside in the United States on an H-1 visa.

This document constitutes proof of such recognition. The original of this document is on file in the Office of Enrollment and Discipline of the U.S. Patent and Trademark Office.

Expires: August 28, 2004

A handwritten signature in black ink, appearing to read "Harry I. Moatz", is written over a horizontal line.

Harry I. Moatz
Director of Enrollment and Discipline

PATENT ABSTRACTS OF JAPAN

1/2

(11)Publication number : 2000-242811

(43)Date of publication of application : 08.09.2000

(51)Int.Cl.

G06T 17/00
G06T 15/70
G06T 15/00
// A63F 13/00

(21)Application number : 11-045535

(71)Applicant : SEGA ENTERP LTD

(22)Date of filing : 23.02.1999

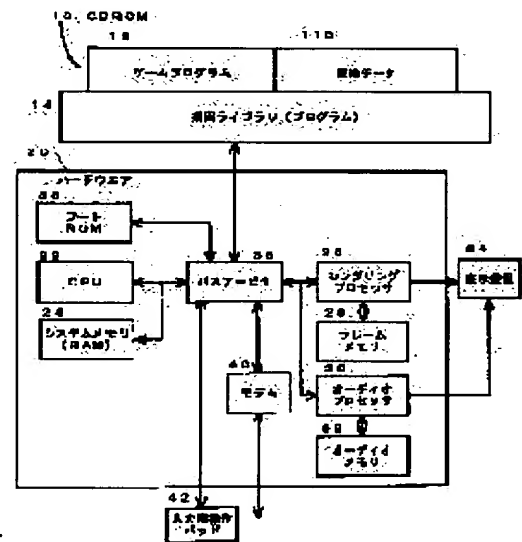
(72)Inventor : SAITO TOMOAKI
ANDO TAKASHI

(54) IMAGE PROCESSING METHOD AND ITS IMAGE PROCESSOR, AND RECORDING MEDIUM WHERE INTEGRATED SHAPED MODEL DATA AND IMAGE PROCESSING PROGRAM USED FOR THE SAME ARE RECORDED

(57)Abstract:

PROBLEM TO BE SOLVED: To make drawable an integrated shaped model capable of actualizing more natural movement of its joints and external surface by providing in real time conversion data for a model in optimum data structure for a drawing library which is recorded on a recording medium for game and which performs image processing in real time.

SOLUTION: In the recording medium 10 for game, a game program 12 which controls the story of the game and the movement of objects including characters in synchronism with the progress of the game, conversion data 110 for a model generated by converting source data by a converter, the drawing library (one kind of program) 14 which performs image processing, etc., are recorded. This recording medium 10 is mounted on the hardware 20 of a game machine and properly read and the game program 12 is executed to execute the drawing library 14 by making use of the conversion data for the model. Consequently, an image is generated in every short time, e.g. frame by frame.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-242811

(P2000-242811A)

(43) 公開日 平成12年9月8日 (2000.9.8)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード*(参考)
G 0 6 T 17/00		G 0 6 F 15/62	3 5 0 A 2 C 0 0 1
15/70		A 6 3 F 9/22	C 5 B 0 5 0
15/00		G 0 6 F 15/62	3 4 0 K 5 B 0 8 0
// A 6 3 F 13/00		15/72	4 5 0 A 9 A 0 0 1

審査請求 未請求 請求項の数28 O L (全 39 頁)

(21) 出願番号 特願平11-45535

(22) 出願日 平成11年2月23日 (1999.2.23)

(71) 出願人 000132471

株式会社セガ・エンタープライゼス
東京都大田区羽田1丁目2番12号

(72) 発明者 斉藤 智昭

東京都大田区羽田1丁目2番12号 株式会
社セガ・エンタープライゼス内

(72) 発明者 安藤 隆

東京都大田区羽田1丁目2番12号 株式会
社セガ・エンタープライゼス内

(74) 代理人 100094525

弁理士 土井 健二 (外1名)

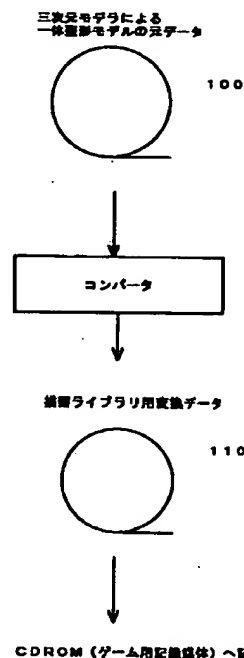
最終頁に続く

(54) 【発明の名称】 画像処理方法、その画像処理装置及びそれに利用する一体整形モデルデータ及び画像処理プログラムを記録した記録媒体

(57) 【要約】 (修正有)

【課題】 一体整形モデルのキャラクタをリアルタイムで描画することができる画像処理方法、一体整形モデルのデータ構造及び画像処理方法を実行するプログラムを記録した記録媒体を提供する。

【解決手段】 画像処理方法は、階層構造のトレース順に複数のモデルに対応する共有頂点バッファを生成する工程と、ゲーム進行データに基づいてモデルの位置を設定したモデルマトリクスデータを生成する工程と、モデルの頂点リストが有する頂点データに対して、モデルマトリクスデータに従うマトリクス演算と、移動後の頂点データにウェイト値を反映するウェイト演算とを行い、当該演算された頂点データを、共有頂点バッファ内の前記頂点IDに従う領域に格納又は加算して共有頂点データを生成する工程と、ポリゴンを、共有頂点データに従ってレンダリングするレンダリング工程とを有する。



【特許請求の範囲】

【請求項1】階層構造でリンクされた複数のモデルを有し、少なくとも第1のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第1の頂点の位置は複数のモデルの位置と当該モデルからのウェイト値に従って影響を受ける一体整形モデルデータを記録した記録媒体において、

前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータと、

前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点IDによって特定された頂点データを有する頂点リストと、

前記モデル毎に形成され、前記複数のモデルの基本姿勢におけるモデル位置を設定したモデルマトリクスデータと、

前記ポリゴンを有するモデル毎に形成され、前記頂点IDを属性データとするポリゴンデータを有するポリゴンリストとを有し、

前記頂点リスト内の頂点データには、少なくとも頂点の位置データと、前記頂点リストが属するモデルからのウェイト値及び当該頂点に対応する前記共有頂点バッファ内の頂点IDとを有することを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

【請求項2】請求項1において、

前記複数のモデルから影響を受ける前記第1の頂点は、前記複数のモデルの個数分だけの分配頂点データを有し、当該分配頂点データは、前記第1の頂点に影響を与えるモデルの前記頂点リストに属することを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

【請求項3】請求項2において、

前記頂点リストは、前記モデルの階層構造に従ってトレースした時に最初に頂点演算が行われるスタートウェイト頂点と、前記頂点演算が最初でも最後でもないミドルウェイト頂点と、前記頂点演算が最後に行われるエンドウェイト頂点とに分けられていることを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

【請求項4】請求項1乃至3のいずれかにおいて、

前記頂点リストは、更に、当該頂点リストが属するモデルからのみ影響を受けるネイティブ頂点のデータを有することを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

【請求項5】請求項1乃至4のいずれかにおいて、

前記頂点データは、前記頂点の座標データと法線データとを有することを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

【請求項6】請求項5において、

前記頂点データは、それが属する頂点リストのモデルのローカル座標系に従うことを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

【請求項7】請求項1乃至4のいずれかにおいて、

前記共有頂点バッファ内の頂点IDは、各モデル毎の頂点のエントリ番号と各モデル毎の頂点数に対応するオフセット値とを有することを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

10 【請求項8】請求項1乃至4のいずれかにおいて、

前記ポリゴンリストは、所定のモデルの前記ポリゴンの描画を指示する描画コマンド又は当該ポリゴンリストが属するモデルのポリゴンの未描画を指示する未描画コマンドを有することを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

【請求項9】階層構造でリンクされた複数のモデルを有し、少なくとも第1のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第1の頂点の位置は複数のモデルの位置と当該モデルからのウェイト値に従って影響を受ける一体整形モデルデータを記録した記録媒体において、

前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータと、

前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点IDによって特定された頂点データを有する頂点リストと、

前記ポリゴンを有するモデル毎に形成され、前記頂点IDを属性データとするポリゴンデータを含むポリゴンリストとを有することを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体。

【請求項10】階層構造でリンクされた複数のモデルを有し、少なくとも第1のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第1の頂点の位置は複数のモデルの位置と当該モデルからのウェイト値に従って影響を受ける一体整形モデルデータを元モデルデータから変換する画像処理方法において、

前記元モデルデータは、前記階層構造でリンクされた複数のモデルデータを有し、前記元モデルデータは、当該モデル内の頂点データを有する元頂点リストと、当該モデルが影響を与える頂点データを有するウェイトリストと、前記頂点リスト内の頂点IDを属性データとする元ポリゴンリストとを有し、

前記画像処理方法は、

前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータを生成する工程と、

前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点IDによって特定された頂点データを有する頂点リストを生成する工程と、

前記ポリゴン有するモデル毎に形成され、前記頂点 ID を属性データとするポリゴンデータを有するポリゴンリストを生成する工程とを有することを特徴とするモデルデータ変換用の画像処理方法。

【請求項 11】請求項 10 において、前記頂点リスト内の頂点データには、少なくとも頂点の位置データと、前記頂点リストが属するモデルからのウエイト値及び当該頂点に対応する前記共有頂点バッファ内の頂点 ID とを有することを特徴とするモデルデータ変換用の画像処理方法。

【請求項 12】請求項 11 において、前記頂点リストを生成する工程は、前記複数のモデルから影響を受ける前記第 1 の頂点に対して、前記複数のモデルの個数分だけの分配頂点を生成し、当該第 1 の頂点に影響を与えるモデルの前記頂点リストに前記分配頂点のデータを分配することを特徴とするモデルデータ変換用の画像処理方法。

【請求項 13】請求項 12 において、前記頂点リストを生成する工程は、前記頂点リスト内に前記頂点データを、前記モデルの階層構造に従ってトレースした時に最初に頂点演算が行われるスタートウエイト頂点と、前記頂点演算が最初でも最後でもないミドルウエイト頂点と、前記頂点演算が最後に行われるエンドウエイト頂点とに分けることを特徴とするモデルデータ変換用の画像処理方法。

【請求項 14】請求項 10 乃至 13 のいずれかにおいて、前記頂点 ID は、各モデル毎の頂点のエントリ番号と各モデル毎の頂点数に対応するオフセット値とを有することを特徴とするモデルデータ変換用の画像処理方法。

【請求項 15】請求項 10 乃至 14 のいずれかにおいて、前記画像処理方法は、更に、前記モデルの階層構造に従うトレース順にモデルの頂点リスト内の頂点データに対するダミー頂点演算を行い、1 つのモデルの頂点リストに対する前記ダミー頂点演算を終了した時に、当該モデル内の頂点に対するウエイト演算が終了しない場合は、当該モデルのポリゴンリスト内に未描画コマンドを発生し、未描画モデル内の頂点に対するウエイト演算が終了した場合は、前記ダミー頂点演算の対象モデルのポリゴンリスト内に前記未描画モデルに対する描画コマンドを発生する描画コマンド生成工程を有することを特徴とするモデルデータ変換用の画像処理方法。

【請求項 16】請求項 10 において、前記画像処理方法は、更に、前記複数のモデルから影響を受ける前記第 1 の頂点を、前記複数のモデルからの影響度が最も大きいモデルからのみ影響を受ける頂点に変換し、当該モデルの前記頂点リストに前記頂点のデータを分配するワンウエイト用頂

点リスト生成工程を有することを特徴とするモデルデータ変換用画像処理方法。

【請求項 17】階層構造でリンクされた複数のモデルを有し、少なくとも第 1 のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第 1 の頂点の位置は複数のモデルの位置と当該モデルからのウエイト値に従って影響を受ける一体整形モデルを描画する画像処理方法において、

前記一体整形モデルのデータが、

10 前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータと、

前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点 ID によって特定された頂点データを有する頂点リストと、

前記ポリゴン有するモデル毎に形成され、前記頂点 ID を属性データとするポリゴンデータを含むポリゴンリストとを有し、

前記画像処理方法は、

20 前記階層構造のトレース順に前記複数のモデルに対応する共有頂点バッファを前記フォーマットデータに従って生成する工程と、

ゲーム進行データに基づいてモデルの位置を設定したモデルマトリクスデータを生成する工程と、

前記モデルの頂点リストが有する頂点データに対して、前記モデルマトリクスデータに従って移動後の頂点データを生成するマトリクス演算と、前記移動後の頂点データに前記モデルからのウエイト値を反映するウエイト演算とを行い、当該演算された頂点データを、前記共有頂点バッファ内の前記頂点 ID に従う領域に格納又は加算して共有頂点データを生成する工程と、

30 前記ポリゴンを、前記共有頂点データに従ってレンダリングするレンダリング工程とを有することを特徴とする画像処理方法。

【請求項 18】請求項 17 において、

前記頂点リスト内の頂点データは、少なくとも頂点の位置データと、前記頂点リストが属するモデルからのウエイト値及び当該頂点に対応する前記共有頂点バッファ内の頂点 ID とを有し、

40 前記共有頂点データを生成する工程において、前記マトリクス演算では前記位置データに前記モデルマトリクスを乗算し、前記ウエイト演算では前記演算された位置データに前記ウエイト値を乗算することを特徴とする画像処理方法。

【請求項 19】請求項 18 において、

前記頂点リストは、前記モデルの階層構造に従ってトレースした時に最初に頂点演算が行われるスタートウエイト頂点と、前記頂点演算が最初でも最後でもないミドルウエイト頂点と、前記頂点演算が最後に行われるエンドウエイト頂点とに分けられ、

前記共有頂点データを生成する工程において、前記スタートウエイト頂点に対する前記演算された頂点データを、前記共有頂点バッファ内の前記頂点IDに従う領域に格納し、前記ミドルウエイト頂点及びエンドウエイト頂点に対する前記演算された頂点データを、前記共有頂点バッファ内の前記頂点IDに従う領域内の頂点データに加算することを特徴とする画像処理方法。

【請求項20】請求項17または18において、前記ポリゴンリストは、所定のモデルの前記ポリゴンの描画を指示する描画コマンド又は当該ポリゴンリストが属するモデルのポリゴンの未描画を指示する未描画コマンドを有し、前記レンダリング工程は、各モデルの前記共有頂点データ生成工程が終了するたびに、当該モデルのポリゴンリストを参照し、前記描画コマンドに従って前記レンダリングを行うことを特徴とする画像処理方法。

【請求項21】請求項20において、更に、前記レンダリングが行われたモデルに対応する前記共有頂点バッファの領域を開放する工程を有することを特徴とする画像処理方法。

【請求項22】請求項17において、前記モデルデータは、更に、前記複数のモデルから影響を受ける前記第1の頂点を、前記複数のモデルからの影響度が最も大きいモデルからのみ影響を受ける頂点に変換し、当該モデルの前記頂点リストに前記頂点のデータを分配したワンウエイト用頂点リストを有し、前記画像処理方法は、所定のモデルに対しては、前記ワンウエイト用頂点リストに従って前記共有頂点データの生成を行うことを特徴とする画像処理方法。

【請求項23】階層構造でリンクされた複数のモデルを有し、少なくとも第1のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第1の頂点の位置は複数のモデルの位置と当該モデルからのウエイト値に従って影響を受ける一体整形モデルを描画する画像処理プログラムを記録した記録媒体において、

前記記録媒体は、前記一体整形モデルのデータを記録し、

当該一体整形モデルのデータが、

前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータと、

前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点IDによって特定された頂点データを有する頂点リストと、

前記ポリゴンを有するモデル毎に形成され、前記頂点IDを属性データとするポリゴンデータを含むポリゴンリストとを有し、

前記画像処理プログラムは、

前記階層構造のトレース順に前記複数のモデルに対応する共有頂点バッファを前記フォーマットデータに従って

生成する手順と、

ゲーム進行データに基づいてモデル間の相対位置関係を設定したモデルマトリクスデータを生成する手順と、

前記モデルの頂点リストが有する頂点データに対して、前記モデルマトリクスデータに従って移動後の頂点データを生成するマトリクス演算と、前記移動後の頂点データに前記モデルからのウエイト値を反映するウエイト演算とを行い、当該演算された頂点データを、前記共有頂点バッファ内の前記頂点IDに従う領域に格納又は加算して共有頂点データを生成する手順と、

前記ポリゴンを、前記共有頂点データに従ってレンダリングするレンダリング手順とをコンピュータに実行させることを特徴とする画像処理プログラムを記録した記録媒体。

【請求項24】請求項23において、

前記頂点リスト内の頂点データは、少なくとも頂点の位置データと、前記頂点リストが属するモデルからのウエイト値及び当該頂点に対応する前記共有頂点バッファ内の頂点IDとを有し、

前記共有頂点データを生成する手順において、前記マトリクス演算では前記位置データに前記モデルマトリクスを乗算し、前記ウエイト演算では前記演算された位置データに前記ウエイト値を乗算することを特徴とする画像処理プログラムを記録した記録媒体。

【請求項25】請求項24において、

前記頂点リストは、前記モデルの階層構造に従ってトレースした時に最初に頂点演算が行われるスタートウエイト頂点と、前記頂点演算が最初でも最後でもないミドルウエイト頂点と、前記頂点演算が最後に行われるエンドウエイト頂点とに分けられ、

前記共有頂点データを生成する手順において、前記スタートウエイト頂点に対する前記演算された頂点データを、前記共有頂点バッファ内の前記頂点IDに従う領域に格納し、前記ミドルウエイト頂点及びエンドウエイト頂点に対する前記演算された頂点データを、前記共有頂点バッファ内の前記頂点IDに従う領域内の頂点データに加算することを特徴とする画像処理プログラムを記録した記録媒体。

【請求項26】請求項23または24において、

前記ポリゴンリストは、所定のモデルの前記ポリゴンの描画を指示する描画コマンド又は当該ポリゴンリストが属するモデルのポリゴンの未描画を指示する未描画コマンドを有し、

前記レンダリング手順は、各モデルの前記共有頂点データ生成手順が終了するたびに、当該モデルのポリゴンリストを参照し、前記描画コマンドに従って前記レンダリングを行うことを特徴とする画像処理プログラムを記録した記録媒体。

【請求項27】請求項26において、

更に、前記レンダリングが行われたモデルに対応する前

記共有頂点バッファの領域を開放する手順を有することを特徴とする画像処理プログラムを記録した記録媒体。

【請求項28】請求項23において、前記モデルデータは、更に、前記複数のモデルから影響を受ける前記第1の頂点を、前記複数のモデルからの影響度が最も大きいモデルからのみ影響を受ける頂点に変換し、当該モデルの前記頂点リストに前記頂点のデータを分配したワンウェイト用頂点リストを有し、前記画像処理プログラムは、所定のモデルに対しては、前記ワンウェイト用頂点 リストに従って前記共有頂点データの生成を行う手順をコンピュータに実行させることを特徴とする画像処理プログラムを記録した記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、リアルタイムで画像処理を行う画像処理方法、リアルタイムで画像処理を行うに適した画像データ構造を生成する画像処理方法、及び当該画像データ構造または画像処理プログラムを記録した記録媒体に関する。

【0002】

【従来の技術】家庭用のビデオゲーム機等において、ゲームの進行に同期してリアルタイムでゲーム内のオブジェクトの動画が作成される。かかるゲーム機における画像処理は、例えばフレーム期間という短い期間内でオペレータからの入力操作信号に応答して、ゲーム内のオブジェクトの位置を移動させ、その移動位置に対応した画像を画像メモリ（フレームメモリ）内に描画する必要がある。

【0003】ゲーム内のオブジェクトのうち、生き物などのキャラクタは、動作の中で手足や首の動きに伴いそれらの関節部分や筋肉の部分も動き、しかも、その関節部分等では他のポリゴンの動きに応じて異なる形状の関節になることが、より自然な動きを表現するためには必要である。或いは、例えば人間の上腕と肩の部分や上腕と腕の部分では、ポリゴンの動きによっては、上腕や肩の筋肉が盛り上がるなどの動きが含まれることが、より自然な動きを表現するためには必要である。

【0004】このような関節部分などに自然な動きを与える描画方法として、関節の両側のモデルに対して接続すべき頂点を指定し、その頂点間に関節用のポリゴンを追加する方法がある。しかし、かかる描画方法では画一的なポリゴンによって関節部分が描画されることになり、関節の角度の変化に対して対応する動きを与えることができず、不自然な画像しか生成できない。

【0005】より自然な動きを描画するために、三次元モデルによりエンベロープと呼ばれる一体整形モデルを利用することが、映画などのリアルタイム処理を必要としないCGムービー（コンピュータ・グラフィックを利用した映画）における画像処理の分野で提案されてい

る。この一体整形モデルは、オブジェクト（キャラクタ）をポリゴンとそれに影響を与える骨部分とで構成したモデルを利用する。即ち、一体整形モデルは、キャラクタの外面であって実際に動いて描画されるポリゴンと、そのポリゴンの位置に一定のウェイトで影響を与える骨部分（bone）とで構成される。そして、ポリゴンの位置に影響を与える骨部分の新たな位置を、オペレータの操作入力に従って求められる位置変換用マトリクスにより演算し、その骨部分からの影響度（ウェイト）に応じてポリゴンの頂点の位置を求める。その求められた頂点に対してポリゴンをレンダリング（描画）することで、複雑な動きをポリゴンに与えることができる。

【0006】図1は、上記のCGムービーにおける一体整形モデルの動きの例を示す図である。図1に示されたモデルは、腕部分1と手の部分2とで構成されるモデルであり、図1（A）は、その基本姿勢の状態を示す。かかる状態から、腕部分1の上腕3がより垂直になり、腕部分1の肘の部分4はより水平方向に曲げられるとすると、図1（B）の如き状態になることが望まれる。即ち、上腕3と肘の部分4を接続する関節部分が自然に曲がった形状になり、かつ上腕3の内側の筋肉3Aが盛り上がるような動きが望まれる。

【0007】図2は、従来のCGムービーにおける一体整形モデルを利用した画像処理を説明する図である。また、図3は、その画像処理手順を示すフローチャート図である。図2には、一体整形モデル（エンベロープ）のモデルのデータ構造620が示される。この例では、モデル1～7の合計7個のモデルが図示されるツリー状の階層構造をなす。そして、各モデルのデータ構造は、モデル4のデータ624とモデル7のデータ627に示される通りである。

【0008】モデルは、表示されるポリゴンを有する場合は、そのポリゴンの構成要素である頂点を有する。従って、モデル4のデータ構造624には、頂点リスト632とその頂点の組み合わせからなるポリゴンリスト635が含まれる。頂点リスト632には、各頂点のモデルのローカル座標系での位置情報と法線データとが含まれる。また、各モデルは、通常その上の階層のモデルに対する相対的な位置を示す情報であるモデルマトリクス631を有する。例えば、モデル4の場合は、モデル3に対する相対的な位置情報として、回転（ローテーション）、移動（トランスレーション）及びサイズ（スケール）を有する。かかる位置情報からなるモデルマトリクス631は、モデル3の座標系からモデル4の座標系に変換する時のマトリクスデータともいえる。

【0009】更に、モデル4のデータ構造624は、モデル4が影響を与える他のモデルの頂点と、その影響度であるウェイト値からなるウェイトリスト633を有する。例えば、図1の例では、肘の部分4のモデルは、上腕3の筋肉3Aの部分に対して所定のウェイト値で影響

を与える。従って、肘部分4のモデルのウェイトリストには、上腕3のモデルのIDとその頂点インデックス（頂点ID）及びそのウェイト値が含まれる。

【0010】仮に、図2のモデル構造の内、モデル4を図1の肘部分4、モデル7を上腕部分3とすると、上腕部分3の動きと肘部分4の動きに影響を受けた結果、上腕部分3のモデルにおける筋肉部分3Aでの頂点の位置が変化し、筋肉部分3Aが盛り上がった画像が描画される。そこで、図3に従って、CGムービーなどで一般的に行われている三次元モデラーの画像処理の手順を説明する。まず、全てのモデルの動きを実現するためのマトリクスを計算し、各モデルのデータ構造に格納する（ステップ602）。そして、ウェイトリスト633を参照し、自分（モデル4）が影響するモデル（モデル7）の頂点インデックス（頂点ID）が示す頂点座標と法線データ（頂点データ）を、モデル7の頂点リスト632から読み出し（ステップ603）、その読み出した頂点データを、基本姿勢時の関係からモデル4のローカル座標系に所属させた場合の頂点データに変換し、その変換した頂点座標と法線データに、モデル4のモデルマトリクスを乗算する（ステップ604）。更に、モデルマトリクスを乗算した頂点座標と法線データに、ウェイトリスト内のウェイト値を乗算した値を、モデル7のデータ構造627内におけるウェイト頂点リスト634内の値に積算する。

【0011】上記の手順603、604、605を、ウェイトリスト全てについて実行し（ステップ606）、更に、全てのモデルについて実行する（ステップ607）。その結果、全てのモデルが持つ実頂点におけるキャラクターの動きに対応する座標変換と、他のモデルからの影響度に応じた位置移動が行われる。そこで、全てのモデルにおけるウェイト頂点リストの法線ベクトルを正規化し、後のレンダリング時における光源処理に使用できる形態にする（ステップ608）。そして、ウェイト頂点リスト内の頂点位置を参照しながら、ポリゴンリストにあるポリゴンの描画（レンダリング）を実行する（ステップ609）。

【0012】上記の演算は、全てのモデルの頂点に対して全てマトリクス演算とウェイト演算を行い、それらの演算が終了してからポリゴンのレンダリング処理を行う。或いは、一つの頂点に注目して、その頂点に対して影響を与えるモデルのマトリクス演算、ウェイト演算を行い、その頂点の新たな頂点座標を求め、最後のレンダリング処理を行う。

【0013】

【発明が解決しようとする課題】しかしながら、上記のCGムービーにおける画像処理方法は、ゲーム機において行われるリアルタイムでの高速画像処理には適用不可能である。即ち、全てのモデルの頂点に対してマトリクス演算とウェイト演算を行う場合は、モデルのデータ構

造内において、異なるモデルの頂点リストやモデルマトリクスを参照しながら、演算を行う必要がある。かかる演算を、メモリ上に分散されたモデルデータの階層構造内をポインタを利用した参照手順を伴って行う場合、ポインタの変更を頻繁に且つ多数回行わなければならない。従って、かかる演算をゲーム機で行おうとすると、ゲーム機内のCPUに設けられたキャッシュメモリの使用効率が悪い。また、一つの頂点に注目してマトリクス演算とウェイト演算を行う場合でも、全ての頂点に対してウェイト演算のためのバッファ領域を確保する必要がある。かかる演算も、スーパーコンピュータ等の大容量のメモリと高速の演算能力を有する場合は別として、簡単な構成のゲーム機で実行すると、利用すべきメモリ領域が膨大になり、リアルタイムでの処理は困難であり、現実的ではない。

【0014】そこで、本発明の目的は、一体整形モデルを利用してリアルタイムで画像処理を行うことができる画像処理方法、その画像処理に利用する一体整形モデルの画像データを生成する画像処理方法、及び当該画像データや画像処理プログラムを記録した記録媒体を提供することにある。

【0015】更に、本発明の目的は、一体整形モデルを利用してリアルタイムで画像処理を可能にする画像データを三次元モデルデータから生成するコンバータプログラムによる画像処理方法を提供することにある。

【0016】更に、本発明の目的は、キャラクターの動きに応じて関節部分などが自然な形に変形することができる画像処理方法、それを実行する画像処理プログラムを記録した記録媒体、その画像処理に適した構造のデータを記録した記録媒体を提供することにある。

【0017】

【課題を解決するための手段】上記の目的を達成するために、本発明は、階層構造でリンクされた複数のモデルを有し、少なくとも第1のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第1の頂点の位置は複数のモデルの位置と当該モデルからのウェイト値に従って影響を受ける一体整形モデルデータを記録した記録媒体において、前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータと、前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点IDによって特定された頂点データを有する頂点リストと、前記モデル毎に形成され、前記複数のモデルの基本姿勢におけるモデル位置を設定したモデルマトリクスデータと、前記ポリゴンを有するモデル毎に形成され、前記頂点IDを属性データとするポリゴンデータを有するポリゴンリストとを有し、前記頂点リスト内の頂点データには、少なくとも頂点の位置データと、前記頂点リストが属するモデルからのウェイト値及び当該頂点に対応する前記共有頂点バッファ内の頂点IDとを有することを特徴とする一

11

体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体を提供する。

【0018】上記の一体整形モデルデータを記録した記録媒体は、ゲーム機器などにおいてリアルタイムでモデルの描画に必要な画像処理を行うことを可能にする。

【0019】更に、本発明は、上記の発明において、前記複数のモデルから影響を受ける前記第1の頂点は、前記複数のモデルの個数分だけの分配頂点データを有し、当該分配頂点データは、前記第1の頂点に影響を与えるモデルの前記頂点リストに属することを特徴とする。

【0020】更に、本発明は、上記の発明において、前記頂点リストは、前記モデルの階層構造に従ってトレースした時に最初に頂点演算が行われるスタートウェイト頂点と、前記頂点演算が最初でも最後でもないミドルウェイト頂点と、前記頂点演算が最後に行われるエンドウェイト頂点とに分けられていることを特徴とする。

【0021】更に、本発明は、上記の発明において、前記頂点リストは、更に、当該頂点リストが属するモデルからのみ影響を受けるネイティブ頂点のデータを有することを特徴とする。

【0022】更に、本発明は、上記の発明において、前記頂点データは、前記頂点の座標データと法線データとを有することを特徴とする。

【0023】更に、本発明は、上記の発明において、前記頂点データは、それが属する頂点リストのモデルのローカル座標系に従うことを特徴とする。

【0024】更に、本発明は、上記の発明において、前記共有頂点バッファ内の頂点IDは、各モデル毎の頂点のエントリ番号と各モデル毎の頂点数に対応するオフセット値とを有することを特徴とする。

【0025】更に、本発明は、上記の発明において、前記ポリゴンリストは、所定のモデルの前記ポリゴンの描画を指示する描画コマンド又は当該ポリゴンリストが属するモデルのポリゴンの未描画を指示する未描画コマンドを有することを特徴とする。

【0026】上記の目的を達成するために、本発明は、階層構造でリンクされた複数のモデルを有し、少なくとも第1のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第1の頂点の位置は複数のモデルの位置と当該モデルからのウェイト値に従って影響を受ける一体整形モデルデータを記録した記録媒体において、前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータと、前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点IDによって特定された頂点データを有する頂点リストと、前記ポリゴンに属するモデル毎に形成され、前記頂点IDを属性データとするポリゴンデータを含むポリゴンリストとを有することを特徴とする一体整形モデルデータを記録したコンピュータ読みとり可能な記録媒体を提供する。

12

【0027】上記の目的を達成するために、本発明は、階層構造でリンクされた複数のモデルを有し、少なくとも第1のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第1の頂点の位置は複数のモデルの位置と当該モデルからのウェイト値に従って影響を受ける一体整形モデルデータを元モデルデータから変換する画像処理方法において、前記元モデルデータは、前記階層構造でリンクされた複数のモデルデータを有し、前記元モデルデータは、当該モデル内の頂点データを有する元頂点リストと、当該モデルが影響を与える頂点データを有するウェイトリストと、前記頂点リスト内の頂点IDを属性データとする元ポリゴンリストとを有し、前記画像処理方法は、前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータを生成する工程と、前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点IDによって特定された頂点データを有する頂点リストを生成する工程と、前記ポリゴンに属するモデル毎に形成され、前記頂点IDを属性データとするポリゴンデータを有するポリゴンリストを生成する工程とを有することを特徴とするモデルデータ変換用の画像処理方法を提供する。

【0028】上記の画像処理方法によれば、リアルタイムでの画像処理を可能にする一体整形モデルの変換データを生成することができる。

【0029】更に、本発明は、上記の画像処理方法において、前記頂点リスト内の頂点データには、少なくとも頂点の位置データと、前記頂点リストが属するモデルからのウェイト値及び当該頂点に対応する前記共有頂点バッファ内の頂点IDとを有することを特徴とする。

【0030】更に、本発明は、上記の画像処理方法において、前記頂点リストを生成する工程は、前記複数のモデルから影響を受ける前記第1の頂点に対して、前記複数のモデルの個数分だけの分配頂点を生成し、当該第1の頂点に影響を与えるモデルの前記頂点リストに前記分配頂点のデータを分配することを特徴とする。

【0031】更に、本発明は、上記の画像処理方法において、前記頂点リストを生成する工程は、前記頂点リスト内に前記頂点データを、前記モデルの階層構造に従ってトレースした時に最初に頂点演算が行われるスタートウェイト頂点と、前記頂点演算が最初でも最後でもないミドルウェイト頂点と、前記頂点演算が最後に行われるエンドウェイト頂点とに分けることを特徴とする。

【0032】更に、本発明は、上記の画像処理方法において、前記頂点IDは、各モデル毎の頂点のエントリ番号と各モデル毎の頂点数に対応するオフセット値とを有することを特徴とする。

【0033】更に、本発明は、上記の画像処理方法において、前記画像処理方法は、更に、前記モデルの階層構造に従うトレース順にモデルの頂点リスト内の頂点デー

タに対するダミー頂点演算を行い、1つのモデルの頂点リストに対する前記ダミー頂点演算を終了した時に、当該モデル内の頂点に対するウェイト演算が終了しない場合は、当該モデルのポリゴンリスト内に未描画コマンドを発生し、未描画モデル内の頂点に対するウェイト演算が終了した場合は、前記ダミー頂点演算の対象モデルのポリゴンリスト内に前記未描画モデルに対する描画コマンドを発生する描画コマンド生成工程を有することを特徴とする。更に、本発明は、上記の画像処理方法において、前記画像処理方法は、更に、前記複数のモデルから影響を受ける前記第1の頂点を、前記複数のモデルからの影響度が最も大きいモデルからのみ影響を受ける頂点に変換し、当該モデルの前記頂点リストに前記頂点のデータを分配するワンウェイト用頂点リスト生成工程を有することを特徴とする。

【0034】上記目的を達成するために、本発明は、階層構造でリンクされた複数のモデルを有し、少なくとも第1のモデルはポリゴンを構成する複数の頂点を有し、少なくとも第1の頂点の位置は複数のモデルの位置と当該モデルからのウェイト値に従って影響を受ける一体整形モデルを描画する画像処理方法において、前記一体整形モデルのデータが、前記複数のモデル内における複数の頂点のデータを、モデル毎に格納する共有頂点バッファのフォーマットデータと、前記頂点に影響を与えるモデル毎に形成され、前記共有頂点バッファ内の頂点IDによって特定された頂点データを有する頂点リストと、前記ポリゴンを有するモデル毎に形成され、前記頂点IDを属性データとするポリゴンデータを含むポリゴンリストとを有し、前記画像処理方法は、前記階層構造のトレース順に前記複数のモデルに対応する共有頂点バッファを前記フォーマットデータに従って生成する工程と、ゲーム進行データに基づいてモデルの位置を設定したモデルマトリクスデータを生成する工程と、前記モデルの頂点リストが有する頂点データに対して、前記モデルマトリクスデータに従って移動後の頂点データを生成するマトリクス演算と、前記移動後の頂点データに前記モデルからのウェイト値を反映するウェイト演算とを行い、当該演算された頂点データを、前記共有頂点バッファ内の前記頂点IDに従う領域に格納又は加算して共有頂点データを生成する工程と、前記ポリゴンを、前記共有頂点データに従ってレンダリングするレンダリング工程とを有することを特徴とする画像処理方法を提供する。

【0035】上記の画像処理方法によれば、一体整形モデルのキャラクターの描画をリアルタイムで行うことができる。

【0036】更に、本発明は、上記の画像処理方法において、前記頂点リスト内の頂点データは、少なくとも頂点の位置データと、前記頂点リストが属するモデルからのウェイト値及び当該頂点に対応する前記共有頂点バッファ内の頂点IDとを有し、前記共有頂点データを生成

する工程において、前記マトリクス演算では前記位置データに前記モデルマトリクスを乗算し、前記ウェイト演算では前記演算された位置データに前記ウェイト値を乗算することを特徴とする。

【0037】更に、本発明は、上記の画像処理方法において、前記頂点リストは、前記モデルの階層構造に従ってトレースした時に最初に頂点演算が行われるスタートウェイト頂点と、前記頂点演算が最初でも最後でもないミドルウェイト頂点と、前記頂点演算が最後に行われるエンドウェイト頂点とに分けられ、前記共有頂点データを生成する工程において、前記スタートウェイト頂点に対する前記演算された頂点データを、前記共有頂点バッファ内の前記頂点IDに従う領域に格納し、前記ミドルウェイト頂点及びエンドウェイト頂点に対する前記演算された頂点データを、前記共有頂点バッファ内の前記頂点IDに従う領域内の頂点データに加算することを特徴とする。

【0038】更に、本発明は、上記の画像処理方法において、前記ポリゴンリストは、所定のモデルの前記ポリゴンの描画を指示する描画コマンド又は当該ポリゴンリストが属するモデルのポリゴンの未描画を指示する未描画コマンドを有し、前記レンダリング工程は、各モデルの前記共有頂点データ生成工程が終了するたびに、当該モデルのポリゴンリストを参照し、前記描画コマンドに従って前記レンダリングを行うことを特徴とする。

【0039】更に、本発明は、上記の画像処理方法において、更に、前記レンダリングが行われたモデルに対応する前記共有頂点バッファの領域を開放する工程を有することを特徴とする。

【0040】更に、本発明は、上記の画像処理方法において、前記モデルデータは、更に、前記複数のモデルから影響を受ける前記第1の頂点を、前記複数のモデルからの影響度が最も大きいモデルからのみ影響を受ける頂点に変換し、当該モデルの前記頂点リストに前記頂点のデータを分配したワンウェイト用頂点リストを有し、前記画像処理方法は、所定のモデルに対しては、前記ワンウェイト用頂点リストに従って前記共有頂点データの生成を行うことを特徴とする。

【0041】上記の目的を達成するために、本発明は、上記した画像処理方法をコンピュータに実行させるプログラムを記録した記録媒体を提供する。この記録媒体によれば、ゲーム機器などにおいて一体整形モデルのキャラクターをリアルタイムで描画処理することができる。

【0042】

【発明の実施の形態】以下、図面を参照して本発明の実施の形態例を説明する。しかしながら、かかる実施の形態例が、本発明の技術的範囲を限定するものではない。

【0043】図4は、本実施の形態例における三次元モデラーを利用して生成される一体整形モデルの例を示す図である。図4の一体整形モデルは、図1に示したよう

な腕の動きを実現するための例である。モデルM1は、実際に表示されるポリゴンを有し、かかるポリゴンは全部で13個の実頂点V101～V113を有する。このモデルM1は、例えば図1における上腕3と肘部分4を含むエンベロープである。また、モデルM1は、自分自身の位置が例えば頂点V101、V107に対して100%影響を与える。従って、モデルM1は、実頂点に対してウェイト値を有する骨モデルでもある。

【0044】モデルM2、M3、M4は、それぞれポリゴンを含まず、従って、実頂点も有しない。但し、モデルM2、M3、M4の動きは、モデルM1内の一部の頂点に対して所定のウェイト値に従って影響を与える。即ち、モデルM2、M3、M4は、実頂点に対してウェイト値を有する骨モデルである。更に、モデル5は、図1の手の部分2に該当するポリゴンを有し、ポリゴンを形成する6個の実頂点V501～V506は、他のモデルからの影響は受けずに、モデルM5自身の位置によって100%の影響を受ける。従って、モデル5は、自分の頂点V501～V506に影響を与える骨モデルでもある。

【0045】図4に示した一体整形モデルは、エンベロープとも呼ばれ、一般にゲームをデザインするデザイナーにより三次元モデラーの利用を介してデザインされる。そして、デザイナーは、各頂点が一つの或いは複数のモデルから所定の割合で影響を受けることをデザインし、ゲームプログラムの実行に伴い各モデルの位置を移動させることで、初期値として決めた影響度（ウェイト値）に応じて頂点の位置を移動させ、キャラクタに所望の動きを与える。

【0046】図5は、図4の一体整形モデルのデータ構成例である。モデルM1の下にモデルM2、M3、M5がつながられ（リンクされ）、更に、モデルM1の下にモデルM4がつながられる。このようなデータのツリー構造は、モデルの階層構造の一例であり、それぞれのプログラムに適した階層構造内に一体整形モデルを構成するモデルがつながられればよい。図5に示したようなツリー構造の場合、一般には、モデルM1、M2、M3、M5、M4の順にトレースされモデルデータへの処理が行われる。かかるトレースの順番もプログラムに適した順番にすることができる。

【0047】図5のデータ構成例において、各モデルは、上層のモデルの座標系に対してそのモデルの座標系がどれだけ移動すべきかを示すモデルマトリクスを有する。即ち、モデルマトリクスは、モデルの位置を設定するデータである。座標系については後に詳述するが、例えば、各モデルはそれぞれのローカル座標系を有する。それとは別に、全てのモデルに共通のグローバル座標系が存在する。そして、例えば、モデルM2の移動後の位置を定義するために、モデルM1のローカル座標系に対するモデルM2のローカル座標系の位置関係として、回

転（ローテーション）、平行移動（トランスレーション）、縮尺（スケール）を含むモデルマトリクスが利用される。従って、モデルM2のローカル座標系の位置座標データや法線データに、このモデルマトリクスを積算するマトリクス演算を行うことにより、モデルM1のローカル座標系の位置座標データや法線データに変換される。また、モデルM1のローカル座標系の位置座標データや法線データに、モデル1のモデルマトリクスを積算すれば、グローバル座標系における位置座標データや法線データを求めることが可能になる。

【0048】このように、各モデル間のモデルマトリクスによって、複数のモデルから構成されるキャラクタに動きが与えられる。通常は、ゲームプログラム中のジオメトリ機能によりかかるキャラクタの動きを示すモデルマトリクスが作成される。また、モデルマトリクスは、グローバル座標系とそのモデルのローカル座標系の位置関係を示してもよい。

【0049】図6は、デザイナーにより設定された各頂点へのモデルからの影響度（ウェイト値）を示す図である。図6の例では、モデルM1の頂点V101、V107は、モデルM1の動きに100%影響を受ける。従って、モデルM1の骨部分の移動に伴って頂点V101、V107の位置も同じ位置関係を保って移動する。また、モデルM1の頂点V102、V103はモデルM2、M3、M4の影響を図示されるウェイト値の割合で受ける。従って、これらの頂点は、モデルM2、M3、M4の移動に伴って図示されるウェイト値の分だけそれらの移動の影響を受ける。例えば、モデルM4が大きく上方に移動すると、その移動の程度の40%分だけこれらの頂点は影響を受ける。例えば、頂点V102、V103を上腕部の筋肉の部分3Aに対応付けると、モデルM4を適宜移動することで、上腕部の筋肉部3Aを適切に盛り上げることが可能になる。

【0050】モデルM1の頂点V104、V105、V108～V111は、図示される通り、モデルM2、M3から図示されるウェイト値の割合で影響を受ける。更に、頂点V106、V112は、モデルM3からのみ影響を受ける。従って、頂点V106、V112は、モデルM3が移動するとそのままの位置関係を維持して移動する。更に、頂点V113は、モデルM4から100%のウェイト値で影響を受ける。

【0051】モデルM5は、モデルM1に接続される部分で、図1の例では手の部分2に対応する。この手の部分2は、特に腕の位置によってその形状が変化する必要はなく、モデルM5の骨部分に100%の影響を受ける。

【0052】図6に示した各頂点のウェイト値は、キャラクタをデザインしたデザイナーによって初期値として設定される。デザイナーは、かかるウェイト値を設定することにより、ゲーム進行中にモデルマトリクスによりモデ

ルの移動を行うことで、各頂点の位置を自在に制御することができる。頂点の位置が決定すれば、その後複数の頂点により決められたポリゴンが、描画部（レンダリング部）により画像メモリ内にレンダリングされる。

【0053】図7は、一体整形モデル（エンベロープ）の移動例を示す図である。この例は、図4に示した基本姿勢にある一体整形モデルが、図1（B）の如く移動した場合に対する一体整形モデルを示す。図1（B）に示したような動きを実現するために、一体整形モデルを構成するモデルM1～M4が次の通り移動する。即ち、モデルM1、M2は、より垂直方向に回転し、モデルM3はより水平方向に回転移動する。それに伴い、モデルM5もモデルM3の先端の位置に整合するように移動する。また、モデルM4は、主に上腕の筋肉の盛り上がり3Aを表現するためのモデルであり、図7に示される通り、より上方により垂直方向に回転移動する。

【0054】その結果、頂点V102、V103はモデルM4の動きに影響を受けて筋肉が盛り上がるように移動し、また頂点V113はモデルM4の動きに連動して移動する。頂点V101、V107は、それぞれモデルM1の動きに連動して移動するだけであり、更に、頂点V104、V105、V108～V111は、モデルM2、M3の動きに影響を受けた位置に移動する。従って、頂点V104、V109、V110などにより形成さえるポリゴンは、腕を曲げた時の自然な肘の関節形状に変化する。また、頂点V112はモデルM3の動きに連動して移動する。更に、モデルM5の各頂点は、それぞれモデルM5の動きにあわせてそのまま移動する。

【0055】そして、移動後の頂点の位置に対して、各頂点の間を埋めるポリゴンがそれぞれレンダリングされることで、図1（B）の如き上腕3をより垂直方向に下げて肘4をより水平方向に曲げ、上腕の筋肉部分3Aが盛り上がり、腕の先に手の部分2が接続された画像が生成される。モデルM2、M3、M4は、それぞれポリゴンを有していないので実頂点はなく、これらのモデルは画像内には表示されない。

【0056】図8は、一体整形モデルにおけるウェイト計算方法を説明する図である。図8には、図4の基本姿勢の状態であった一体整形モデルが、図7の移動後の状態になる場合について、特に頂点V102、V103とそれに影響を与えるモデルM2、M3、M4の関係が示される。また、図9は、一体整形モデルにおけるモデルの移動計算（マトリクス計算）と頂点のウェイト計算の手順を示すフローチャート図である。

【0057】図8（A）は基本姿勢の状態のモデルM2、M3、M4を示す。すでに述べた通り、モデルM2、M3、M4は、それぞれのローカル座標系CDM2、CDM3、CDM4を有する。また、全てのモデルに共通のグローバル座標系CDGBも設定される。図4に示された一体整形モデルにおいて、頂点V102、V

103は、モデルM2、M3、M4から影響（ウェイト値）を受けるようにデザイナーにより設定される。即ち、図6に示したようなウェイト値があらかじめ初期値として設定される。そこで、ウェイト計算を行うために、この頂点V102、V103は、それぞれその頂点に影響を与えるモデルM2、M3、M4に属する3つの頂点（分配頂点）に分けられ、それぞれの対応するモデルに分配される（図9中S1）。例えば、頂点V102は、分配頂点V102（M2）、V102（M3）、V102（M4）に分けられ、それぞれの影響を与えるモデルM2、M3、M4に分配される。頂点V103についても、同様に3つの分配頂点に分けられ、影響を与えるモデルに分配される。

【0058】その結果、モデルM2には、頂点V102（M2）とV103（M2）が属することになる。同様にして、モデルM3には、頂点V102（M3）とV103（M3）が属し、モデルM4には、V102（M4）とV103（M4）が属する。そして、各モデルに属する頂点の基本姿勢状態での位置座標は、各モデルのローカル座標系に変換される（図9中S2）。例えば、頂点V102（M2）と頂点V103（M2）の絶対座標は、モデルM2のモデルマトリクスに従ってモデルM2のローカル座標系CDM2に変換される。それ以外の頂点についても同様に属するモデルのローカル座標系に変換される。この結果、各頂点は、それらが属するモデルと一体に移動することになる。

【0059】次に、図8（B）は、各モデルM2、M3、M4がそれぞれ図7の如く移動した状態を示す。各モデルの移動は、それぞれのモデルマトリクスを設定することにより制御される（図9中S3）。例えば、モデルM2のローカル座標系CDM2は、図示しないモデルM1のローカル座標系に対してモデルM2のモデルマトリクスに対応する回転、平行移動、拡大・縮小した位置に移動する。また、モデルM2に属する頂点V102（M2）とV103（M2）は、基本姿勢の状態のモデルM2に対する位置を保ちながら、モデルM2の移動に従って移動する。即ち、頂点V102（M2）とV103（M2）の位置は、モデルM2のローカル座標系の中では何ら移動することはない。従って、モデルM2が移動することに伴う頂点V102（M2）とV103（M2）の位置も、モデルM2のモデルマトリクスを利用したマトリクス計算をすることにより、求めることができる（図9中S4）。

【0060】モデルM3に属する頂点V102（M3）とV103（M3）も、モデルM4に属する頂点V102（M4）とV103（M4）も同様にして、それぞれのモデルの移動に伴い移動し、それらの位置は、それぞれのモデルのモデルマトリクスを利用したマトリクス計算をすることで求められる。

【0061】次に、各モデルに分配した頂点の例えばグ

ローカル座標系における位置が求められると、各モデルが頂点に与えるウェイト値（影響度）に応じたウェイト演算が行われる（図9中S5）。例えば、図6のウェイト値によれば、モデルM2、M3、M4からのウェイト値は、両頂点共に、30%、30%、40%である。従って、頂点V102について説明すれば、頂点V102（M2）の頂点座標値に0.3を乗算し、頂点V102（M3）の頂点座標値に0.3を乗算し、更に頂点V102（M4）の頂点座標値に0.4を乗算し、それらを積算（加算）することにより、各モデルからの影響度（ウェイト値）に応じた新たな頂点V102の位置座標が求められる。即ち、図8（B）中に二重丸で示した頂点V102が、3つの分配頂点V102（M2）、V102（M3）、V102（M4）から30%、30%、40%のウェイト値により求められた実頂点の位置である。3つの分配頂点V102（M2）、V102（M3）、V102（M4）のグローバル座標系におけるX座標値を、X2、X3、X4とすると、二重丸で示した実頂点V102のX座標値X102は、

$X102 = 0.3 \times X2 + 0.3 \times X3 + 0.4 \times X4$
で求められる。かかる計算が、ウェイト計算である。

【0062】このようにして求められた二重丸で示したウェイト計算後の頂点の座標値や法線データを利用して、複数の頂点からなるポリゴンのレンダリングが行われる。例えば、複数の頂点で結ばれたポリゴン内のピクセルの色データは、ラスタスキャン法により頂点座標に対するピクセル座標の内分比を求め、頂点データやテクスチャデータなどから、その内分比を利用した補間演算により求められる。上記の頂点座標に加えて、通常は頂点の法線ベクトルについてのデータも利用される。上記と同じウェイト演算により二重丸の頂点における法線データが求められ、かかる法線データを利用して光源に対する拡散光の演算などが行われる。これらのレンダリング方法は、本発明の話題とするところではなく、一般的なレンダリング方法が利用されることができる。

【0063】以上が、3Dモデラーを利用した一体整形モデル（エンベロープ）によるキャラクタの動きを実現する方法である。上記説明した通り、一体整形モデルを利用した場合、各モデルは自分の実頂点データに加えて、影響を与える頂点のデータも有する必要があり、各モデルの位置をモデルマトリクスを利用した演算により求める工程、同様にモデルマトリクスを利用して影響を与える頂点の位置を求める工程、それら頂点に対してウェイト演算により影響度を考慮した新たな位置と法線データを求める工程、そうして求めた頂点データを利用して各モデルのポリゴンのレンダリングを行う工程などが必要である。このような工程を伴う一体整形モデルに対する処理を、ゲーム機器のようなスーパーコンピュータと比較すると簡単な演算機能や少ないメモリ容量しか有しないハードウェアを利用して、リアルタイムで行う必

要がある。

【0064】本実施の形態例では、一体整形モデルの元モデルデータから、上記のデータ処理の一部を予め行って得られた変換データを利用する。この変換データは、ゲーム機器によるリアルタイムでの演算、処理に適した形態を有し、かかる変換データが、キャラクタデータとしてゲーム機器用の記録媒体、例えばCDROMやゲームカセット、に記録される。

【0065】図10は、本実施の形態例における一体整形モデルの元データと描画ライブラリ用の変換データの関係を示す図である。本実施の形態例によれば、CGムービーなどにおいて利用される三次元モデラーにより生成された一体整形モデルの元データ100が、本発明により提供される新規なコンバータプログラムにより、描画ライブラリ用の変換データ110に変換される。この変換データ110は、ゲーム進行中にダイナミックに変化するデータを含まず、ゲーム機が実行するゲームプログラムを記録した記録媒体内の描画ライブラリ（描画処理プログラム）によって、ゲーム進行中のキャラクタ等のダイナミックな動きを演算するために最適化されたデータフォーマットを有する。従って、この変換データ110は、あらかじめコンバータプログラムによって元データから変換されて生成され、ゲーム用の記録媒体、例えばCDROM内に記録される。

【0066】図11は、ゲーム用の記録媒体（例えばCDROM）内の構成と、ゲーム機のハードウェアとの関係を示す図である。図11に示されるとおり、ゲーム用の記録媒体10内には、ゲームの進行に同期して、ゲームのストーリーやキャラクタを含むオブジェクトの動きを制御するゲームプログラム12と、コンバータにより元データから変換されて生成されたモデルの変換データ110と、画像処理を行う描画ライブラリ（プログラムの一種）14などが記録される。かかるゲーム用記録媒体10は、ゲーム機のハードウェア20に装着され、適宜読み出されて、そのゲームプログラムが実行され、モデルの変換データを利用して描画ライブラリ12が実行され、その結果、例えばフレーム毎の様に短時間毎の画像が生成される。

【0067】図11には、ゲーム機のハードウェア20の構成が示される。ゲーム機のハードウェア20は、演算処理を行うCPU22と、ゲーム用記録媒体10のプログラムやモデルの変換データを一時的に読み込み、また演算処理のバッファメモリとして利用されるシステムメモリ（RAM）24とがバスにより接続される。また、バスアービタ38を介して、描画ライブラリによって生成される描画コマンドに従ってレンダリングを行うレンダリングプロセッサ26と、そのプロセッサ26により描画された画像信号が記録されるフレームメモリ28と、音声データを生成するオーディオプロセッサ30と、その生成した音声データを記録するオーディオメモ

り32とが接続される。フレームメモリ28に記録された画像データ及びオーディオメモリ32に記録された音声データが、モニタテレビ等の表示装置34から表示され、出力される。また、ゲーム機器20内のバスアービタ38はインターフェースとしての機能も有し、ゲームの入力用操作パッド42が入力手段として接続され、モデム40を介して外部の通信回線と接続される。また、バスアービタ38は、ブートROM36とも接続され、電源投入時のブート動作が、このブートROMのデータを実行することにより行われる。更に、ゲーム機器20は、ゲーム用の記録媒体10、例えばCDROMやゲームカートリッジ、をバスアービタ38を介して接続し、外部記録装置として利用する。

【0068】ゲーム用記録媒体10に記録されているモデルの変換データ110は、描画ライブラリ14による画像処理の負担を軽減できるデータフォーマットに変換されている。このデータフォーマットについては、後述するが、モデルの変換データ110は、描画ライブラリ14に最適なデータ構造であるので、ゲーム進行中に移動するキャラクタの一体整形モデルの画像処理がリアルタイムで可能になる。

【0069】図12は、三次元モデラによる一体整形モデルのデータから変換データを作成するまでの概略的フローチャート図である。ゲームの作成には、通常ゲームのストーリーなどを創作する企画と、ゲーム内のキャラクタなどのオブジェクトをデザインするデザイナと、ゲームプログラムを創作するプログラマとにより行われる。三次元モデラは、より自然な動きを持つキャラクタを提供するために上記のデザイナにより利用され、エンベロープと呼ばれる一体整形モデルの元データが生成される(S10)。この一体整形モデルについては、前述した通りであり、例えば、従来例として図2に示した通り、モデル毎に、モデルマトリクスと、頂点データと、ウェイトリストなどを有する。モデルマトリクスは、モデルの階層構造(ツリー構造)における上層のモデルに対する回転、平行移動、拡大・縮小についてのマトリクスデータである。また、頂点データは、実頂点に対応するデータであり、頂点の位置座標データと法線ベクトルについての法線データを少なくとも含む。それ以外に、頂点データには、例えば不透明度を示す α 値、テキストチャデータに対応するテキスト座標などが含まれる。また、ウェイトリストは、そのモデルが影響を与える実頂点のID(モデルIDとそれに属する頂点ID)と、ウェイト値を少なくとも有する。このウェイトリストを展開したのが、図6に示されたデータである。

【0070】次に、コンバータプログラムにより、一体整形モデルの元データから描画ライブラリ用のデータフォーマットに対応した変換データが生成される(S12)。この変換データの詳細は後に説明するが、概略的に述べると、実頂点をウェイトなしのネイティブ頂点

と、ウェイト計算が必要なスタートウェイト頂点、ミドルウェイト頂点及びエンドウェイト頂点とからなる4つのグループに分ける。スタートウェイト頂点は、ウェイト計算において最初にウェイト値の乗算が行われる頂点であり、ミドルウェイト頂点は、ウェイト計算においてその次にウェイト値の乗算が行われる頂点であり、エンドウェイト頂点は、ウェイト計算で最後にウェイト値の乗算が行われる頂点である。ウェイト計算の順番は、通常モデルの階層構造(ツリー構造)のトレース順に従って決定される。各グループに分配された頂点は、それぞれの頂点が属するモデル毎にソーティングされる。

【0071】そして、実頂点を重複することなく示すネイティブ頂点とスタートウェイト頂点からなる共有頂点バッファのフォーマットが生成され、その共有頂点バッファに並べられる実頂点に対して、モデルの属性にかかわらず統一的な頂点IDが与えられる。共有頂点バッファ内のネイティブ頂点とスタートウェイト頂点とで構成される実頂点のリストは、それら実頂点が属するモデル毎の固まりになる。各モデル毎の共有頂点リストは、更に、それらの実頂点に影響を与えるモデル毎にソーティングされ、モデル毎の順番で格納される。かかる共有頂点バッファに並べられたモデル毎の実頂点リストに対して、統一的な頂点IDが与えられ、後のウェイト計算での参照手順を簡素化する。上記の様に生成された共有頂点バッファは、従来のウェイト計算を伴わないモデルの実頂点リストと同等の単純なデータ構造を有する。

【0072】共有頂点バッファのフォーマットに加えて、4つのグループに分けられた頂点は、それらの頂点に対して影響を与えるモデルに分配される。そして、各モデル毎に分配された頂点からなる頂点リストが生成される。この頂点リストは、上記4つのグループ、ネイティブ、スタートウェイト、ミドルウェイト及びエンドウェイトにソーティングされ、格納される。また、この頂点リスト内の頂点データには、モデル内のローカル座標に従う頂点座標データと、法線データ、及び必要に応じて影響を与える実頂点の共有頂点バッファ内の頂点IDを引用したウェイトデータを含む。即ち、スタートウェイト、ミドルウェイト、及びエンドウェイトに属する頂点データには、上記のウェイトデータが含まれ、ネイティブに属する頂点データには含まれない。

【0073】このモデル毎の頂点リストには、そのモデルからのみ影響を受ける(ワンウェイトの)ネイティブ頂点と、他のモデルと共にそのモデルが影響を与える(マルチウェイトの)スタートウェイト頂点、ミドルウェイト頂点、エンドウェイト頂点とが、一元的にまとめられる。かかる頂点リストと共有頂点バッファにより、描画ライブラリは、リアルタイムでの画像処理を行う。

【0074】モデルの変換データには、更に、ポリゴンリストが含まれる。このポリゴンリストは、そのポリゴンを構成する複数の頂点データを有する。但し、このポ

リゴンリスト内の頂点データは、共有頂点バッファ内の実頂点に与えられた頂点IDを属性データとする。更に、ポリゴンリストには、好ましくはそのモデルのレンダリング（描画）を行うべきか否かを示す描画コマンドも含まれる。この描画コマンドは、コンバータにより生成され、描画ライブラリにより参照される。

【0075】以上の通り、コンバータにより生成される変換データ110には、実頂点のリストを意味する共有頂点バッファのフォーマットと、各モデルに分配された頂点リストと、各モデルのポリゴンリストとを有する。図9に戻って説明すると、コンバータによる変換手順には、図9の実頂点のモデルへの分配（S1）と、そのモデル内のローカル座標系への変換（S2）とを含み、更に、描画ライブラリにより実行されるモデルマトリクスを利用した移動計算（マトリクス演算）（S3、S4）とその後のウェイト値を利用したウェイト演算（S5）を行うのに最適な変換データの生成が行われる。この変換データについては、後に図4、図7の例を元に再度説明する。

【0076】さて、変換データ110の生成と共に、上記したプログラムによりゲームプログラムが作成される（S14）。そして、図11に示したとおり、ゲームプログラム12と、モデルの変換データ110と、描画ライブラリ14とが、ゲーム用記録媒体10内に記録される（S16）。

【0077】図13は、ゲームにおけるフローチャート図である。ゲームを行う場合、まずゲーム用記録媒体10をゲーム機器に装着する（S20）。ゲーム装置は、記録媒体のゲームプログラムに従ってゲームを実行し、キャラクタを構成するモデルの移動用のモデルマトリクスを生成する（S22）。移動用のモデルマトリクスは、前述した通り、上層のモデルに対する移動、回転、拡大・縮小などのパラメータからなるマトリクスであり、モデル或いはそのポリゴンの頂点座標にモデルマトリクスを乗算することにより新たな移動後の座標が求められる。

【0078】次に、ゲーム機器は、描画ライブラリを実行する（S24）。モデルの変換データ110は、全てのモデルの実頂点について統一したIDを与えた共有頂点バッファのフォーマット、モデル毎に分配された頂点リスト、モデル毎のポリゴンリストを有する。モデル毎の頂点リストは、後述する通り他のモデルから影響を受けない頂点（ネイティブ頂点）と、他のモデルからも影響を受ける頂点（スタートウェイト、ミドルウェイト、エンドウェイト）とからなり、更に他のモデルから影響を受ける頂点は、影響を与える当該他のモデル毎にソートされている。また、モデル毎の頂点リストは、共有頂点バッファ内の統一された頂点IDに従って、影響を与える他の頂点IDを引用する。そして、モデル毎のポリゴンリストには、モデルを構成するポリゴンの頂点ID

を属性データとする頂点データが含まれ、またウェイト計算が終了したモデルについての描画コマンドや、ウェイト計算が終了しないモデルについての未描画コマンドが含まれる。更に、共有頂点バッファは、描画ライブラリが画像処理を行う段階で、ゲーム機器内のシステムメモリ（RAM）内に展開される。

【0079】描画ライブラリの概略の手順では、まず、ゲームプログラムにより作成された移動用マトリクスを各モデルの頂点の座標データに対して乗算することで、新たな移動後の頂点の座標データを求める。この工程は、図9のステップS4に対応する。更に、その頂点データに基づき、モデルの階層構造のトレース順に従って且つモデル内の頂点リストの順番にウェイト計算を行い、影響を与える頂点に対応する共有頂点バッファ内の領域にその計算結果を書込みまたは積算（加算）する。この工程は、図9のステップS5に対応する。次に、描画ライブラリは、各モデルのポリゴンリスト内の描画コマンドに従って、共有頂点バッファ内に格納された頂点座標及び法線データを参照してモデル内のポリゴンの描画（レンダリング）を行い、ポリゴン内のピクセルの色データなどを含む画像データを、フレームバッファ28内に記録する。

【0080】フレームバッファ26内に描画された画像データに従って、表示装置34にその画像を表示する（S26）。フレーム毎にこの描画された画像を表示することにより、オペレータの操作入力信号に応答して動くキャラクタを含む画像を表示することができる。しかも、三次元モデラーにより生成された一体整形モデルによるモデルが利用されているので、キャラクタの関節部分はより自然な動きをなし、キャラクタの筋肉も盛り上がり等の動きも表現することができる。以上の工程S20～S26が、フレーム期間毎に繰り返される。

【0081】図14及び図15は、コンバータの手順を示すフローチャート図である。本実施の形態例のコンバータは、ゲーム用の記録媒体に格納される描画コマンドと違ってリアルタイムでの処理は要求されない。このフローチャートの工程S30、S32、S33が、モデルの元データから共有頂点バッファのフォーマットの作成と、モデル毎の頂点リストの作成を行う段階である。また、フローチャートの工程S36～S60は、描画ライブラリが実行する画像処理のシュミレーション演算段階であり、このシュミレーション演算を行うことにより、モデルの階層構造をトレースした時の各モデルの描画（レンダリング）を行うタイミングを検出し、未描画コマンドや描画コマンドをモデルのポリゴンリスト内に挿入することができる。これにより作成される共有頂点バッファのフォーマットと頂点リストは、描画ライブラリが画像処理を行うに適したデータ形式であり、更に、シュミレーション演算により生成される未描画コマンド又は描画コマンド付きのポリゴンリストは、描画ライブラ

りによる各モデルの描画タイミングの管理負担を軽減させる。

【0082】図14に示された処理手順を有するコンバータプログラムは、ゲーム機器よりもより高速演算機能を有しより大容量のメモリを有する汎用のコンピュータによって実行される。この汎用コンピュータのハードディスク等の外部記録媒体内に、コンバータプログラムが記録され、元データに対して処理手順が実行され、変換データが生成される。

【0083】さて、図9において説明した一体整形モデルの描画における移動計算（マトリクス計算）とウエイト計算を行うためには、実頂点をそれに影響を与えるモデルに分配し（S1）、その分配したモデルのローカル座標系に頂点の位置の座標データや法線データを変換する（S2）必要がある。本実施の形態例のコンバータは、上記の2つの手順S1、S2に加えて、実頂点を分配されるモデルの数だけ発生して、単一のモデルの位置に依存する（ウエイトを持たない）ネイティブ頂点、複数のモデルから影響を受ける頂点（ウエイトを持つ頂点）のうちモデル階層をトレースした時に最初にウエイト計算を行う必要があるスタートウエイト頂点、同様に2番目以降にウエイト計算を行う必要があるミドルウエイト頂点、同様に最後にウエイト計算を行う必要があるエンドウエイト頂点の4つのグループに分ける（S30）。

【0084】更に、コンバータは、それら4グループの頂点の内、ネイティブ頂点とスタート頂点を実頂点に対応するので、それらの頂点グループをそれらが属するモデル毎に並び替えて、共有頂点バッファのフォーマットを作成する（S32）。この共有頂点バッファに展開されるこれらの頂点に対して、統一的な通し番号が新たな頂点IDとして与えられる。この新たな頂点IDは、種々の形態の番号が採用可能である。例えば、各モデル毎の頂点のエントリ番号とそのモデル内の頂点数に対応するオフセット値を利用することにより、共有頂点バッファのフォーマットデータを形成することができ、同時に頂点IDにも利用することができる。

【0085】更に、コンバータは、各モデルに分配された4グループの頂点に対して、それら頂点に影響を与えるモデル毎に並び変えて、各モデルの頂点リストを作成する（S34）。

【0086】上記の処理S30、S32、S34を、図4、図6のモデル例を使ってより具体的に説明すると、以下の通りである。図16は、より詳細な実頂点のモデルへの分配と頂点リストの生成の手順を示すフローチャート図である。また、図17は、元データから変換データへの手順を示す図である。そして、図18は、具体的モデル例に適用した時の元データの頂点リストと共有頂点リストを示し、図19は、具体的モデル例に適用した時の頂点リストを示す図である。適宜図18、19を参

照しながら、図16、17に示された手順を説明する。

【0087】図17の左側若しくは図18の左側に示される通り、元データのモデルM1、M5のデータには、頂点リスト（図17中107）が含まれる。この元データのモデルデータは、3Dモデラーにより生成されるデータであり、例えば図2に示した様に、各モデルにはモデルマトリクス（図17中109）を、ポリゴン有するモデルは頂点リスト（図17中107）及びポリゴンリスト（図17中106）を、そして、頂点に影響を与えるモデルはウエイトリスト（図17中108）を有する。図18の左側の元頂点リストは、上記のポリゴン有するモデルM1、M5の頂点リストに対応する。従って、ここに示された頂点を実頂点であり、頂点リストには、少なくとも頂点の位置座標データと法線データとが含まれる。

【0088】最初に、コンバータは、基本姿勢における一体整形モデルの実頂点の位置座標及び法線データのグローバル座標を求める（S62）。この演算は、例えばモデルM1、M5のモデルマトリクスを、それぞれのモデルの実頂点データに乗算することで行われる。この結果、図4に示された実頂点V101～V113及びV501～V506のデータがグローバル座標系に変換される。

【0089】次に、元のモデルデータの実頂点を、モデル階層のトレース順に従って、ウエイトの有無に基づく4つのグループに分ける（S62、図17内の工程121～124参照）。図18の具体例で説明すると、モデル階層のトレース順は、モデルM1、M2、M3、M5、M4になる。また、モデルM1の頂点V101は、モデルM1の位置に100%依存する頂点であり、ウエイト計算を必要としないウエイトを持たないネイティブ頂点グループに属する。モデルM1の頂点V102は、図6のウエイト表に示され、図8で示した通り、モデルM2、M3、M4にそれぞれ30%、30%、40%で影響を受けるので、モデルM2に影響を受ける頂点V102（M2）はスタートウエイトグループに、モデルM3に影響を受ける頂点V102（M3）はミドルウエイトグループに、そして、モデルM4に影響を受ける頂点V102（M4）はエンドウエイトグループに分類される。

【0090】同様に、モデルM1の頂点V103についても、モデルM2に影響を受ける頂点V103（M2）はスタートウエイトグループに、モデルM3に影響を受ける頂点V103（M3）はミドルウエイトグループに、そして、モデルM4に影響を受ける頂点V103（M4）はエンドウエイトグループに分類される。モデルM1の頂点V104は、モデルM2に影響を受ける頂点V104（M2）をスタートウエイトグループに、モデルM3に影響を受ける頂点V104（M4）をエンドウエイトグループに分類される。図18に示される通

り、頂点V105、V108～V112も、同様にスタートウエイトグループとエンドウエイトグループとに分類される。

【0091】更に、図18に示される通り、モデルM1の頂点V106は、モデルM3に100%影響を受けるので、ネイティブグループに分類され、頂点V107は、モデルM1に100%影響を受け、頂点V113は、モデルM4に100%影響を受けるので、それぞれの頂点V107(M1)、V113(M4)は、ネイティブグループに分類される。

【0092】また、モデルM5の実頂点V501～V506は、全て別のモデルからのウエイトを受けないでモデルM5の位置に依存するので、それぞれ頂点V501(M5)～V506(M5)は、ネイティブグループに分類され、モデルM5に分配される。以上の通り、実頂点は、4つの頂点グループに分類される。次に、コンバータは、上記4つのグループの頂点を、それら頂点に影響を与えるモデル毎にソーティングする(S66、図17中の工程125～128)。例えば、図18の具体例で説明すると、モデルM1のネイティブグループの4つの頂点V101(M1)、V106(M3)、V107(M1)、V113(M4)は、それら頂点に影響を与えるモデルM1、M3、M4毎にソーティングされ、例えば、V101(M1)、V107(M1)、V106(M3)、V113(M4)の順番に並び替えられる。また、モデルM1のスタートウエイトグループの9つの頂点V102(M2)～V105(M2)及びV108(M2)～V112(M2)は、全てモデルM2から影響を受けるのでそのままの順番になる。更に、モデルM1のミドルウエイトグループに属する2つの頂点V102(M3)、V103(M3)は、そのままの順番になる。そして、エンドウエイトグループに属する9つの頂点V102(M4)、V103(M4)、V104(M3)、V105(M3)、V108(M3)～V112(M3)は、それらの頂点に影響を与えるモデルM3、M4毎に並び替えられる。モデルM5の6つの頂点は、全てモデルM5から影響を受けるので、その並びは変わらない。

【0093】次に、コンバータは、上記4つのグループの内、ネイティブグループとスタートウエイトグループの頂点に対して、共有頂点バッファのフォーマットを生成する(S68、図17中の工程129)。かかる共有頂点バッファは、実質的には実頂点リストを格納するバッファであり、第1に実頂点が属するモデル毎にまとめられ、第2にその中でネイティブグループとスタートウエイトグループ毎にまとめられ、第3にそれらの頂点に影響を与えるモデル毎にまとめられる。

【0094】図18の例で示すと、モデルM1についての共有頂点バッファ内のリストは、ネイティブグループにおいて、影響を与えるモデルM1に対して頂点V10

1、V107を、モデルM3に対して頂点V106を、そしてモデルM4に対して頂点V113を有する。同様に、スタートウエイトグループにおいて、影響を与えるモデルM2に対して9つの頂点V102～V112を有する。スタートウエイトグループにおいて、モデルM1、M3、M4に対する頂点は存在しない。

【0095】そして、共有頂点バッファ内に割り当てられる頂点に対して、新たな通し番号vx0～vx12を新たな頂点IDとして与える。この頂点IDの与え方は、図17中工程129に説明される通り、共有頂点バッファ上で各モデルの頂点数からなるオフセットを先頭とする通し番号をエントリ番号とすることもできる。例えば、モデルM1については、先頭アドレスをADD(M1)と与え、頂点V102の領域は、それから5番目のエントリ番号になるということを示す頂点IDとして、ADD(M1)+4なるIDが設定される。そして、モデルM1内の頂点数は13個であるので、モデルM1に対するオフセット値は13になる。従って、次のモデルM5の先頭アドレスADD(M5)は、モデルM1のオフセット値13が利用され、それを先頭とするエントリ番号0～6がモデルM5の共有頂点に与えられる。

【0096】このようなオフセット値とエントリ番号を利用することで、各モデルの共有頂点リストをバッファ上に展開する時、メモリ領域内のバッファ上での各頂点の領域を示すアドレスを簡単に設定することができる。この点については、後で詳述する。また、かかるオフセット値とエントリ番号が、ここでいう共有頂点バッファのフォーマット情報になる。

【0097】上記した共有頂点リストは、各モデルの実頂点リストと実質的に同じであり、かかるリストをメモリ内に展開して共有頂点バッファを作成することで、ウエイト演算におけるウエイト値と座標又は法線データの乗算値の積算(加算)を共有頂点バッファに対して行うことができる。しかも、各頂点に新たな頂点IDを与えて、後述する頂点リストやポリゴンリスト内においてもこの頂点IDを利用することで、描画ライブラリによる演算時の頂点の参照IDを簡素化することができる。

【0098】新たな頂点IDが与えられたことに対応して、各モデルが有するポリゴンリスト内の頂点IDも、新たな頂点IDに変換される(S68)。そうすることにより、モデルのウエイト演算が終了した後のポリゴンの描画(レンダリング)を行う際に、共有頂点バッファ内のデータの参照を容易にすることができ、リアルタイムでの処理が要求される描画ライブラリによる描画工程(レンダリング工程)を高速化することができる。

【0099】図8で説明した通り、生成された各頂点は、その頂点に影響を与えるモデルに分配される必要がある。従って、コンバータは、4種類の頂点を、その頂点に影響を与えるモデルにそれぞれ分配する(S70)。ここで分配とは、具体的にいえば、頂点をそれに

影響を与えるモデルの頂点リストに含ませ、更に、その頂点の座標データと法線データをモデルのローカル座標系に変換することを意味する。ローカル座標系への変換は、工程S62で求めた各頂点のグローバル座標系のデータに、モデルの位置を示すモデルマトリックスの逆マトリックスを乗算することで行われる。

【0100】コンバータは、上記分配された頂点を、分配先のモデルの頂点リストに登録する(S72)。この頂点リストは、4つのグループ毎にまとめられる。そして、その登録された頂点の参照IDには、共有頂点バッファのフォーマットに従う新たな頂点ID(例えばオフセット値とエントリ番号)を使用する。その結果、頂点リスト上の頂点に対してウェイト演算を行う時、演算結果を格納する共有頂点バッファの領域へのアクセスを容易にすることができる。

【0101】図18、図19の具体例で説明すると、図18の4グループの頂点に対し、それに影響を与えるモデル、つまり図中括弧内に示したモデル(V101(M1))の場合はモデルM1)の頂点リストに分配される。その結果、図19に示される頂点リストが、各モデル毎に生成される。例えば、モデルM1の頂点リストには、ネイティブの頂点V101(M1)とV107(M1)の座標データVERTと、法線データNORMとが並べられる。そして、それぞれの頂点の参照IDは、図中「頂点ID(vx0, 2)」で示される。即ち、「頂点ID(vx0, 2)」は、図18の共有頂点バッファ内の頂点IDがvx0の領域から2つの頂点領域を示す。別の参照IDの与え方としては、例えば、頂点リスト内の頂点データに、新たな頂点IDであるvx0、vx1を属性データとして与えることもできる。

【0102】図19の例に示される通り、モデルM2の頂点リストは、ネイティブの頂点は存在せず、スタートウェイトの9つの頂点が存在する。スタートウェイトの頂点の場合は、頂点データに座標データVERT、法線データNORMに加えて、ウェイト値WTと格納すべき実頂点のID(図19の例ではvx4、vx5)を有するウェイトデータNFtagSWを有する。この実頂点IDは、共有頂点バッファの各頂点ブロックのオフセット値とそのブロック内のエントリ番号とによっても特定される。従って、ウェイトデータNFtagSWには、そのエントリ値を頂点IDとして利用することもできる。

【0103】図19に示したモデルM3の頂点リストの場合は、ネイティブの頂点として頂点V106(M3)の頂点データと、ミドルウェイトの頂点として頂点V102(M3)とV103(M3)の頂点データ(座標データ、法線データ及びウェイトデータ)と、エンドウェイトの頂点として7個の頂点データ(座標データ、法線データ及びウェイトデータ)とが含まれる。また、モデルM4の頂点リストの場合は、ネイティブの頂点が1個、エンドウェイトの頂点が2個含まれる。

【0104】以上の工程S62~S72が、全てのモデルに対して行われた結果、図17に示される通り、各モデルデータ115として、新たな頂点IDによるポリゴンリスト116と、4つのグループ毎に分けられた頂点リスト117が生成される。具体例は、図19に示した通りである。更に、図18に示した通りの、共有頂点リストが生成され、それに対応する共有頂点バッファのフォーマット情報(例えば、オフセット値とエントリ番号)が生成される。その結果、描画ライブラリは、実頂点のデータを格納する共有頂点バッファを作成することができ、各モデルに分配された頂点リストに従って、ネイティブ頂点用の演算、スタートウェイト頂点用の演算、ミドルウェイト頂点用の演算、そしてエンドウェイト用の演算を順番にまとめて行うことができ、更に、共有頂点バッファ内に書き込まれたウェイト演算後の頂点データを利用して、新たな頂点IDによるポリゴンリストを参照して描画を行うこともできる。

【0105】図14に戻り、コンバータは、上記の共有頂点バッファのフォーマット情報、モデル毎の頂点リストとポリゴンリストの作成に加えて、各モデルの描画(レンダリング)タイミングをシュミレーションして、ポリゴンリスト内に描画コマンドまたは未描画コマンドを加える。そのために、モデル内の実頂点に影響を与える頂点のウェイト計算が終了するタイミングを検出する。

【0106】そのタイミングを検出するために、ポリゴン有するモデル毎にウェイト計算回数を示すカウンタ値を決定する(S36)。このカウンタ値は、各モデルにおいて4つのグループに分けられた頂点の数に等しい。即ち、モデルM1のカウンタ値は、23になり、モデルM5のカウンタ値は6になる。このカウンタ値は、別の言い方をすれば、描画ライブラリが共有頂点バッファ内に書き込み又は積算(加算)を行う回数を意味し、各モデルにおいてこのカウンタ値のウェイト計算が終了すると、そのモデルの頂点に影響を与えるモデルのウェイト計算が終了したことを意味し、実頂点のデータが確定し描画(レンダリング)を行う準備ができたことを意味する。

【0107】描画ライブラリによる描画シュミレーションでは、各モデルの頂点リストの頂点データに対して順番にマトリクス計算とウェイト計算を行い共有頂点バッファ内の頂点領域内に書き込み又は積算を行い(S42)、書き込んだ又は積算した頂点が属するモデルのカウンタを+1する(S43)。図19の例で説明すれば、まず、モデルM1の共有頂点バッファ領域をメモリ内に確保し、モデルM1の頂点リストについて、頂点V101(M1)の座標データVERTと法線データNORMに対し、モデルM1のモデルマトリックスを乗算して新たな位置での座標データと法線データを求め、ウェイト値1、0(100%)を乗算し、図18に示した共有頂点バッ

ファ内のvx0の領域にそれらのデータを書き込む。この頂点V101(M1)はネイティブグループであり、モデルM1に100%影響を受けるから、ウエイト値は1.0であり、ウエイト値を乗算してもデータに変更はない。従って、ネイティブグループの頂点については、ウエイト計算(乗算)を行うことなく共有頂点バッファ内にデータを書き込んでもよい。次に、頂点V107(M1)についても同様の計算を行い、共有頂点バッファ内のvx1の領域に求めたデータを書き込む。その結果、モデルM1のカウント値は2になる。

【0108】上記工程S42、S44が、各モデルの頂点リストの順で繰り返される。そして、モデルの全ての頂点が終了すると(S46)、そのモデルに対するウエイト計算が終了したか否かを、モデルのカウント値が設定値と同じか否かにより判定する(S48)。図19のモデルM1の例でいえば、2つの頂点V101(M1)と頂点V107(M1)のウエイト計算が終了しても、カウント値は2であり、設定値の24に達していないので、モデルM1の描画を行うことはできない。従来の複数のポリゴンからなるモデルでは、そのポリゴンを構成する頂点の位置(座標)と法線は、モデルの位置が決まれば画一的に確定していた。即ち、モデルのモデルマトリクスが求められ、座標データや法線データにそのモデルマトリクスを積算すると、新たな座標データと法線データが確定する。従って、その時点でそのモデルの描画(レンダリング)を行うことができる。即ち、基本的にはマトリクス演算が終われば、そのモデルの描画を行うことはできる。

【0109】しかし、一体整形モデル(エンベロープ)の場合は、ほかのモデルからの影響を受けるので、それらの影響についても全てのウエイト演算が終了しないとそのモデルの描画を行うことはできない。

【0110】そこで、本実施の形態例では、モデルのウエイト計算が終了していない場合は、未描画コマンドDRAWCP(ADD)をそのモデルのポリゴンリストに発生し、図21で後述する未描画テーブルに登録し、次のモデルの共有頂点バッファ領域を確保する(S50)。上記未描画コマンドの属性データADDは、ポリゴンリストの番地を示す。図19の例でいえば、モデルM1のポリゴンリストの先頭に、未描画コマンドDRAWCP(ADD)が書き込まれる。この結果、モデルM1のポリゴンリストは完成する。尚、DRAWCPは、DRAW cash polygonの略称である。

【0111】一方、モデルのウエイト計算が終了している場合は、描画コマンドDRAWCP(ADD)をそのモデルのポリゴンリストに加える(S52)。そして、共有頂点バッファの領域を開放する。或いは、描画コマンドを加えることなく、ポリゴンリスト内に未描画コマンドが記載されていなければ、即そのモデルの描画(レンダリング)を実行してもよいというルールにすることもでき

る。即ち、モデルの頂点リストに対するマトリクス計算とウエイト計算が終了すれば描画(レンダリング)を行うことをデフォルトにすることもできる。その場合は、モデルのポリゴンリスト内に上記の描画コマンドは加えられず、未描画コマンドの存在が意味を持つことになる。

【0112】モデルの頂点リストに対する計算が終了すると、そのモデルの頂点リスト、ポリゴンリスト等が出力される(S54)。ポリゴンリスト内には、未描画コマンドに対する未描画テーブルの登録IDや描画コマンドに対する未描画テーブルの登録ID等が含まれる。

【0113】モデルの頂点リスト内の頂点に対する演算が全て終了しなくても、未描画モデルのウエイト計算の終了を、モデル毎のカウント値を参照することで検出する必要がある(S58)。そして、ウエイト計算が終了した未描画モデルに対して、現在処理中のモデル内のポリゴンリストに、描画コマンドDRAWDP(ADD)を発生する。その結果、描画可能状態になったモデルに対する共通頂点バッファの領域が解放される(S60)。尚、DRAWDPとは、DRAW draw polygonの略称である。

【0114】上記のシュミレーション演算が、全てのモデルの頂点リストに対して行われる。

【0115】より具体的に理解するために、図18、19の実例で説明する。図20は、その場合に生成されるポリゴンリスト例であり、図21は未描画テーブルの例である。まず、コンバータは、モデルM1の頂点リストについて、マトリクス演算とウエイト演算を行い、その結果を共有頂点バッファ内の対応する領域に書き込む。そして、頂点リストの全ての頂点データに対する演算が終了しても、ウエイト演算は終了していないので、モデルM1に対するポリゴンリスト145の先頭に、未描画コマンドDRAWCP(0)を登録する。この時、未描画テーブル143内には、このコマンドの属性データの「0」に対するポリゴンリスト先頭番地ADD1の対応が登録される。

【0116】次に、モデルM2の頂点リストについても、同様にマトリクス演算とウエイト演算を行い、共有頂点バッファの対応する領域に書き込み又は積算(加算)を行う。そして、モデルM2の頂点リストの頂点全ての演算が終了した時点で、モデルM1の頂点に対するウエイト演算は終了していないので、次のモデルM3について同様の処理を行う。モデルM4の頂点リストの処理が終了すると、モデルM1の実頂点に対するウエイト演算は終了しモデルM1の描画が可能な状態になる。そこで、モデルM4のポリゴンリスト147の先頭に、未描画状態にあったモデルM1の描画コマンドDRAWDP(0)が登録される。

【0117】モデルM5の頂点リストは、図19に示されないが、全ての頂点がネイティブであるので、モデルM5の頂点リストのマトリクス演算とウエイト演算が終

了した時点で描画可能状態にあるので、モデルM5のポリゴンリスト146内には、モデルM5の描画コマンドDRAWが登録される。或いは、未描画コマンドの登録がなければ描画処理を行うことがデフォルトで設定される場合は、かかるモデルM5の描画コマンドDRAWの登録は必要がない。

【0118】いずれのポリゴンリスト145、146、147にも、ポリゴン名PGとそれを構成する頂点IDを含む、頂点IDは、図20の例では、共有頂点バッファにおいて新たに与えた統一的な頂点IDである $vx0, vx1$ が利用される。共有頂点バッファが生成された時のエントリ番号とオフセット値を利用した実際のアドレスを頂点IDにして、ポリゴンリストを形成することもできる。その例については、後で詳述する。

【0119】図21の未描画テーブル143には、モデルM1のID=0に対して、そのモデルのポリゴンリストの先頭番地ADD1が対応付けされる。未描画テーブル143は、必ずしも必要なく、未描画コマンドDRAWCPの属性データに直接ポリゴンリストの先頭番地ADD1を与えることでもよい。

【0120】以上説明した通り、コンバータは、三次元モデラーにより生成された一体整形モデルの元データから、共有頂点バッファのフォーマット情報と、各モデル毎の頂点リストとポリゴンリスト等を有する変換データを、非リアルタイムで生成する。この変換データは、ゲームプログラムや描画ライブラリと共にゲーム用の記録媒体、例えばCDROM内に記録される。

【0121】次に、共有頂点バッファの生成と解放に伴う、頂点IDについて説明する。図22は、頂点IDを説明するための図である。図22の例では、頂点IDを説明する都合上、上記してきたモデルM1～M5に、その後トレースされるポリゴン付きのモデルM6が加えられている。図22(A)は、モデルの頂点をネイティブ、スタートウエイト、ミドルウエイト、エンドウエイトに分類し、ネイティブとスタートウエイトの頂点から頂点バッファのフォーマットが形成された状態を示す。この状態では、全ての実頂点に通し番号 $vx0 \sim vx(h+i+j-1)$ が与えられる。但し、 h, i, j はそれぞれモデルM1、M5、M6のオフセット値とする。そこで、コンバータによるシュミレーション演算を行うことにより、図22(B)に示される様に、共有頂点バッファの領域が生成され、解放されたとする。

【0122】上記してきたモデルM1～M5では、モデルM1とM5が実頂点を有する。そして、モデル階層のトレース順では、モデルM1、M2、M3、M5、M4で頂点リストに対する演算が行われる。従って、共有頂点バッファ領域の先頭番地がADDmとすると、モデルM1の共有頂点バッファの領域が番地ADDm以降に最初に生成される(図22(B-1))。次に、モデルM5の共有頂点バッファ領域が、モデルM1の領域の後に生成され

る(図22(B-2))。この時点では、モデルM1の頂点のアドレスは、先頭番地ADDmとエントリ値 $0 \sim (h-1)$ により特定される。また、モデルM5の頂点のアドレスは、先頭番地ADDmに加えてモデルM1のオフセット値 h とモデルM5のエントリ値 $0 \sim i$ によって特定される。従って、この時点では、モデルM1とM5の共有頂点のアドレスは、先頭番地ADDmから、 $0 \sim (h+i-1)$ の連続番号で特定される。このアドレスが頂点IDに利用される。

【0123】即ち、この時点で、モデルM1とモデルM2の描画が行われ、その後モデルM1、M5の共有頂点バッファ領域は開放される。従って、各モデルの頂点リストの頂点IDと、モデルM1、M5のポリゴンリストの頂点IDを、全てこのアドレスにすることができる。そうすることにより、描画ライブラリの処理効率を高くすることができる。

【0124】次に、モデルM1、M5の描画が終了すると、それらの共有頂点バッファ領域はもはや必要になる。そこで、次に描画すべきモデルM6の共有頂点バッファ領域は、再度先頭番地ADDmからエントリ番号 $0 \sim (j-1)$ に生成される。従って、このアドレスが、モデルM6等の頂点リストやポリゴンリスト内の頂点IDとして利用される。つまり、共有頂点バッファ領域の生成と維持を必要最小限に止めるために、異なるモデルの共有頂点バッファ領域のアドレスが重複することになる。しかし、モデルの階層構造のトレース順に頂点リストを参照し、ポリゴンリストを参照するので、そのようなアドレスの重複に不都合は生じない。

【0125】別の変形例として、頂点リストやポリゴンリストに統一的な頂点ID $vx0 \sim vx(h+i+j-1)$ を利用し、コンバータのシュミレーション演算により、共有頂点バッファ領域のアドレスと統一頂点IDとの対応テーブルを生成することもできる。その場合は、描画ライブラリは、頂点リストとポリゴンリストの頂点IDに対するバッファ領域のアドレスを、その対応テーブルを参照することで知ることができる。

【0126】図23は、具体的な頂点リストの別の例を示す図である。この例は、上記の共有頂点バッファ領域のアドレスを頂点IDとして積極的に利用した場合の例である。この頂点リスト200は、頂点リスト開始の記述201と終了の記述203の間に、4グループ毎の頂点データの固まり(チャンク)有する。202は、頂点データチャンクの終了を示す記述である。即ち、頂点リスト200は、ウエイトを持たないネイティブ頂点のチャンク218、スタートウエイトの頂点のチャンク228、ミドルウエイトの頂点のチャンク238、及びエンドウエイトの頂点のチャンク248を有する。各チャンクには、ウエイトなしの頂点グループのデータチャンク名を示す記述211と、ウエイトありの頂点グループのデータチャンク名を示す記述221とを有する。それら

の記述には、ウェイト計算継続中のフラグ212とチャンクサイズ213と、スタート、ミドル、エンドの区別表示222、232、242とを含む。

【0127】更に、各チャンクには、共有頂点バッファの頂点リストブロックのオフセット値214とそのチャンク内の頂点データの数215を示す記述も含まれる。頂点データには、図19に示したのと同様に、座標データ216と法線データ217と、ウェイトデータとを有する。ウェイトデータには、頂点の共有頂点バッファ上でのエントリ番号226とウェイト値(%)227とを有する。オフセット値214とエントリ番号226とにより、共有頂点バッファ内の実頂点の領域(頂点ID)を特定することができる。

【0128】次に、描画ライブラリの手順について説明する。図24は、描画ライブラリの手順のフローチャート図である。図13で説明した通り、ゲーム進行中は、オペレータからの操作データに応答してゲームプログラムがキャラクタを構成しているモデルの移動用のマトリクスを生成する(S22)。そこで、描画ライブラリがポリゴンの描画を行う(S24)。

【0129】描画ライブラリの手順は、コンバータのシミュレーション演算に類似する。描画ライブラリは、変換データを利用してリアルタイムでポリゴンの描画を行う。描画ライブラリは、モデル階層構造のトレース順に従い、各モデルの頂点リスト順に、マトリクス演算により移動後の頂点の座標データと法線データを求める(S80)。かかる求められた座標データと法線データは、変換データにモデルマトリクスを乗算することにより求められ、グローバル座標系に変換される。このマトリクス演算に伴い、そのモデルの共有頂点バッファ領域をメモリ内に確保する(S78)。かかる共有頂点バッファ領域は、変換データ内のフォーマット情報により必要なだけ確保することができる。そして、マトリクス演算を行った頂点が、ネイティブグループに分類された頂点の場合は(S82)、ウェイト演算では、単に100%(1.0)を乗算するだけであるので、実際には、マトリクス演算により求めた座標データと法線データを、対応する共有頂点バッファの領域に記録する(S84)。

【0130】また、マトリクス演算を行った頂点が、ウェイトありの頂点、即ち、スタートウェイト、ミドルウェイト或いはエンドウェイトの頂点の場合は(S86)、ウェイトデータを利用してウェイト計算を行い、対応する共有頂点バッファにその計算結果の座標データ及び法線データを書き込むか積算(加算)する。スタートウェイトの頂点の場合は、最初に登場した頂点であるので、計算結果は共有頂点バッファ内に書き込まれるが、ミドルウェイト又はエンドウェイトの頂点の場合は、すでに書き込まれている値に、その計算結果を積算(加算)する。そして、計算結果を記録したモデルのカ

ウンタを+1する(S90)。

【0131】上記のマトリクス演算とウェイト演算とが、モデルの頂点リスト内のすべての頂点データに対して行われる(S92)。全ての頂点IDが統一されたIDを利用し、その頂点IDと共有頂点バッファの頂点データ領域の参照インデックスとが対応しているので、このウェイト演算は、非常に高速に行うことができる。更に、頂点リストは、モデル毎に分けられている。従って、マトリクス演算において、モデル内の頂点に対しては同じモデルマトリクスを使用することができる。このことは、描画ライブラリによるマトリクス演算において、CPU内に設定されるモデルマトリクスを頻繁に変更する必要がないことを意味する。従って、マトリクス演算効率が高くなり、リアルタイムでの描画を可能にする。

【0132】更に、頂点リストは、頂点グループ毎にまとめられているので、まとめられた頂点に対するウェイト演算における手順を、均一化することができる。即ち、ネイティブの頂点であれば、マトリクス演算結果をそのまま共有頂点バッファに書き込み、スタートウェイトの頂点であれば、マトリクス演算してウェイト演算した結果を共有頂点バッファに書き込み、ミドルウェイトとエンドウェイトの頂点は、両演算結果を共有頂点バッファ内のデータに積算(加算)する。このように手順を均一化することができるので、描画ライブラリの演算効率は高くなり、リアルタイムでの演算を可能にする。

【0133】描画ライブラリは、一つのモデルの頂点リストに対する演算が終了すると、そのモデルのポリゴンリストを参照する。ポリゴンリスト内に未描画コマンドDRAWCPが記載されている場合(S94)は、そのモデルの描画(レンダリング)を行うことができないので、未描画テーブルに登録する(S96)。また、ポリゴンリスト内に描画コマンドDRAWDPが記載されている場合は(S98)、未描画モデルについて、未描画テーブルに登録された番地を参照して、ポリゴンリストの順にポリゴンの描画(レンダリング)を行う(S100)。かかるポリゴンのレンダリングは、共有頂点バッファ内に格納された演算結果のグローバル座標系における座標データと法線データを利用して行われる。従って、そのモデルの描画が終了すると、そのモデルに対する共有頂点バッファの領域は不要になり、解放される。解放とは、必要があれば別の共有頂点バッファ領域に割り当てられる状態にすることを意味する。また、ポリゴンリスト内にそのモデルの描画を指示する描画コマンドDRAWが記載される場合は、自分のモデルの描画(レンダリング)し、それに対応する共有頂点バッファ領域を開放する(S102)。

【0134】なお、デフォルトの手順として、モデル内の頂点リストの演算が終了したら、未描画コマンドが存在しない限り自動的にそのモデルのポリゴンリストに従

ってポリゴンの描画を行わせることもできる。更に、描画（レンダリング）を行う前に、頂点計算が終了した法線データについて正規化を必要に応じて行う（S93）。法線データは、レンダリング工程において光源からの光のベクトルと正規化された法線ベクトルとの内積値（ $\cos \theta$ 、 θ は両ベクトル間の角度）を求める時に利用される。この内積値に応じてポリゴン内のピクセルにおける拡散光等の処理が行われる。従って、レンダリング工程における演算によっては、予め法線ベクトルの正規化（スカラー値を1にする処理）を行うことが必要になる。

【0135】描画ライブラリによる描画（レンダリング）工程において、全てのモデルの共有頂点バッファ領域をメモリ内に確保する必要はない。必要なウェイト演算が終了するまでは、モデルに対応する共有頂点バッファ領域を確保しておく必要がある。しかし、一旦全てのウェイト演算が終了して、そのモデルを描画できる状態になると、予めポリゴンリスト内に登録されている描画コマンドに従ってモデル内のポリゴンの描画（レンダリング）を行うことができる。描画が終了すれば、もはやその共有頂点バッファ領域内の頂点データは不要になるので、係る領域を開放して、他のモデルの共有頂点バッファ領域に提供することができる。従って、メモリ内の共有頂点バッファ領域の容量を平均的に小さくすることができる。かかる点は、比較的少ない容量のメモリしか利用できないゲーム機器にとっては大きなメリットになる。

【0136】以上の様に、コンバータによって予め変換テーブルが作成されているので、リアルタイムでのレンダリングが要求される描画ライブラリは、上記の通り非常に効率よくマトリクス演算やウェイト演算を行うことができる。更に、メモリ内に確保すべき実頂点のデータ領域（共有頂点バッファ領域）の容量を小さくすることができる。従って、描画ライブラリは、ゲーム機器を利用してリアルタイムでオペレータからの操作に回答したキャラクタの画像を作成することができる。

【0137】上記の実施の形態例では、一体整形モデルにおいて、実頂点が複数のモデルから影響を受けるマルチウェイトのデータについて、コンバータによる変換データと描画ライブラリの処理について説明した。かかるマルチウェイト方式による一体整形モデルを利用することで、ポリゴン間を接続する関節部分や表面の筋肉の形状をより自然な形で描画することが可能になる。

【0138】しかしながら、ゲーム内のシーンによっては、マルチウェイトの一体整形モデルからなるキャラクタであっても、例えば視点から遠隔の位置にある場合は、複雑なマルチウェイトに従う頂点演算を行う必要がない場合もある。視点から遠隔に位置するキャラクタには、視点から近い位置でスクリーンの中央部に位置するキャラクタのようなより高級な画像を必要としないから

である。そのような場合は、演算処理に時間とハードウェア資源を必要とするマルチウェイトに従う頂点演算ではなく、ワンウェイトに従う頂点演算にすることで、画面全体の画像処理時間とハードウェア資源を節約することができる。ワンウェイトとは、一つのモデルからのみ影響を受ける頂点、つまりネイティブグループに属する頂点、のみからなるモデルを意味する。

【0139】図25は、マルチウェイトの一体整形モデルのデータからワンウェイトの頂点リストを得るコンバータの手順を示すフローチャート図である。本フローチャートは、図16に示したマルチウェイトのデータに対する頂点のモデルへの分配と頂点リストの生成のフローチャートに一部変更を加えている。

【0140】工程S62での、基本姿勢における一体整形モデルの実頂点の座標と法線データをグローバル座標系に変換することは、マルチウェイトの場合と同じである。次に、コンバータは、元のモデルデータの実頂点を、モデル階層のトレース順に従って、ウェイトの有無に基づくグループに分ける（S64）。即ち、マルチウェイトの場合の、ネイティブグループとそれ以外のグループとに分ける。そして、コンバータは、複数モデルからのウェイト（影響）を有する頂点のうち、最大のウェイトを与えるモデルのネイティブの頂点にする（S65）。例えば、図8の例で説明すると、3つのモデルM2、M3、M4からのウェイトを有する頂点V102は、モデルM4からのウェイト値が40%と最大であるので、3つの頂点V102（M2）、V102（M3）、V102（M4）のうち、頂点V102（M4）のみをモデルM4に属する頂点にし、残りのウェイト値が小さい頂点V102（M2）、V102（M3）は無視する。その結果、頂点V102は、モデルM4から100%の影響を受けるネイティブの頂点になる。

【0141】工程S65の処理により、全ての頂点は1つのモデルから影響を受けるシングルウェイトの頂点になる。その後の処理は、図16のマルチウェイの場合と同じである。即ち、頂点を、それに影響を与えるモデル毎にソーティングし（S66）、全ての頂点に対して統一した番号を新たな頂点IDとして与え、各モデルのポリゴンリストに反映する（S68）。或いは、マルチウェイトの場合と同様に、共有頂点バッファのオフセット値とエントリー番号による頂点IDを利用してもよい。

【0142】更に、コンバータは、全ての頂点をその頂点に影響を与えるモデル（頂点が属するモデル）に分配し、頂点データをモデル内のローカル座標系に変換する（S70）。そして、分配先のモデルの頂点リストに、それら分配した頂点を登録する（S72）。シングルウェイトであるので、各モデルの頂点リストには、マルチウェイトの様にネイティブグループ、スタートウェイトグループなどのグループは存在せず、全てネイティブの頂点になる。むしろ、頂点リストでの実頂点の引用は、

共有頂点バッファのフォーマットに従う頂点IDになる。

【0143】即ち、シングルウェイトの場合は、共有頂点バッファ領域には、各モデル毎にそのモデルのポリゴンを構成する実頂点のリストが形成され、各モデルの頂点リストには、そのモデルが影響を与えるネイティブ頂点のリストが形成される。従って、かかるシングルウェイトの変換データを利用すると、描画ライブラリは頂点演算の回数を1つの頂点に対して1回にすることができ、描画工程を簡単にし、高速描画処理を可能にする。

【0144】シングルウェイトの変換データは、コンバータによりマルチウェイト形式の全てのモデルに対して生成されることが好ましい。従って、コンバータは、マルチウェイト形式のモデルに対しては、図16に従ってマルチウェイト形式の変換データと、図25に従ってシングルウェイト形式の変換データとを生成する。そして、描画ライブラリは、ゲームの進行中に、描画すべきキャラクタの位置が、例えば視点位置から遠い位置にある場合や、画面の中央部以外の位置にある場合、更に、例えば半透明ポリゴンの背後に位置する場合等、精密なモデルの動作を要求されない状況にある場合は、シングルウェイト形式の変換データを利用して描画処理を行う。その結果、画面中央部のもっとも目立つ位置にあるキャラクタにのみマルチウェイトの変換データを利用して描画処理を行う。その結果、画面全体に必要な描画処理を軽くすることができる。

【0145】

【発明の効果】以上、本発明によれば、ゲーム（またはシミュレーション）用記録媒体に記録されリアルタイムで画像処理を行う描画ライブラリに最適なデータ構造のモデルの変換データを提供することができるので、より自然な関節や外面の動きを実現できる一体整形モデルを、リアルタイムに描画することが可能になる。

【0146】また、本発明によれば、三次元モデラなどにより生成される一体型モデルのデータを、ゲーム（またはシミュレーション）用記録媒体に記録されリアルタイムで画像処理を行う描画ライブラリに最適なデータ構造のモデルのデータに変換する画像処理方法（コンバータ）を提供するので、三次元モデラを利用した一体整形モデル形式のモデルデータを、ゲーム機器（またはシミュレーション）に利用することができ、より自然なオブジェクトの動きをゲームにおいて提供することができる。

【0147】更に、本発明によれば、変換データを利用して一体整形モデルの描画を効率よく行う画像処理方法またはそれをコンピュータに実行させるプログラムを記録した記録媒体を提供することができるので、ゲーム機器において、リアルタイムで一体整形モデルによるキャラクタを描画することができ、より自然な動きのキャラクタの画像を形成することができる。

【図面の簡単な説明】

【図1】CGムービーにおける一体整形モデルの動きの例を示す図である。

【図2】従来のCGムービーにおける一体整形モデルを利用した画像処理を説明する図である。

【図3】図2の画像処理手順を示すフローチャート図である。

【図4】本実施の形態例における三次元モデラを利用して生成される一体整形モデルの例を示す図である。

【図5】図4の一体整形モデルのデータ構成例である。

【図6】デザイナーにより設定された各頂点へのモデルからの影響度（ウェイト値）を示す図である。

【図7】一体整形モデル（エンベロープ）の移動例を示す図である。

【図8】一体整形モデルにおけるウェイト計算方法を説明する図である。

【図9】一体整形モデルにおけるモデルの移動計算と頂点のウェイト計算の手順を示すフローチャート図である。

【図10】本実施の形態例における一体整形モデルの元データと描画ライブラリ用の変換データの関係を示す図である。

【図11】ゲーム用の記録媒体内の構成と、ゲーム機のハードウェアとの関係を示す図である。

【図12】三次元モデラによる一体整形モデルのデータから変換データを作成するまでの概略的フローチャート図である。

【図13】ゲームにおけるフローチャート図である。

【図14】コンバータの手順を示すフローチャート図である。

【図15】コンバータの手順を示すフローチャート図である。

【図16】詳細な実頂点のモデルへの分配と頂点リストの生成の手順を示すフローチャート図である。

【図17】元データから変換データへの手順を示す図である。

【図18】具体的モデル例に適用した時の元データの頂点リストと共有頂点リストを示す図である。

【図19】具体的モデル例に適用した時の頂点リストを示す図である。

【図20】ポリゴンリスト例を示す図である。

【図21】未描画テーブルの例を示す図である。

【図22】共有頂点バッファの展開と解放の例を示す図である。

【図23】具体的な頂点リストの別の例を示す図である。

【図24】未描画テーブルの例を示す図である。

【図25】描画ライブラリの手順のフローチャート図である。

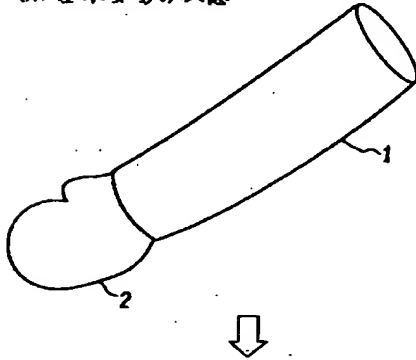
50 【符号の説明】

41
 100 一体整形モデルの元データ
 110 一体整形モデルの変換データ
 10 ゲーム用記録媒体、CDROM
 12 ゲームプログラム

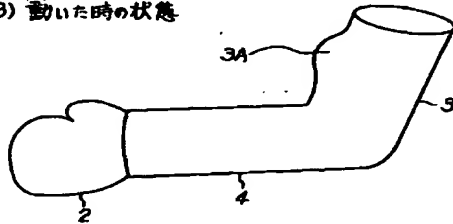
42
 * 14 描画ライブラリ
 20 ゲーム機器
 M1~M6 モデル
 * vx 頂点ID

【図1】

一体整形モデル(エンベロープ)の動きの例
 (A) 基本姿勢の状態

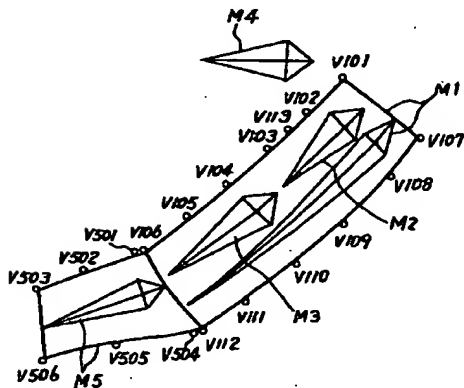


(B) 動いた時の状態



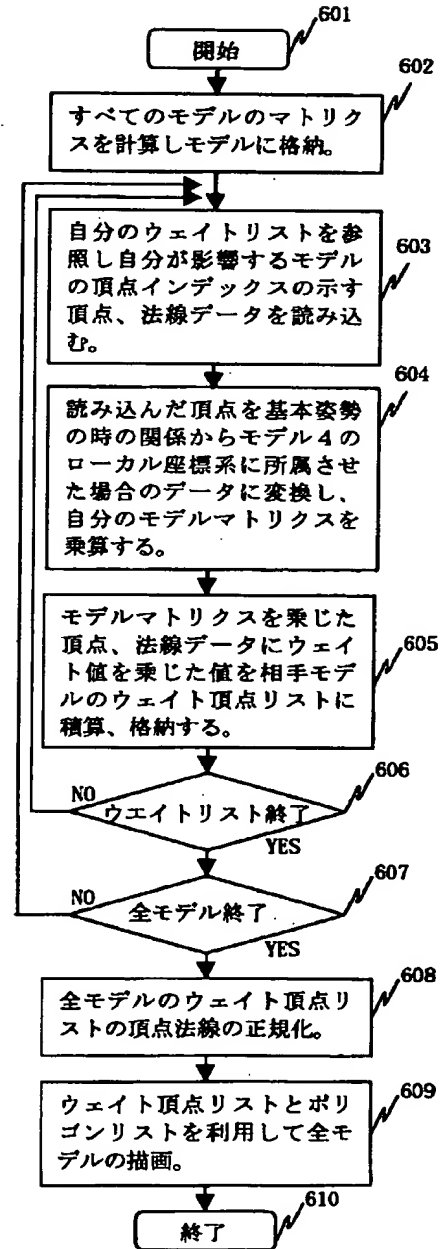
【図4】

一体整形モデル(エンベロープ)の例



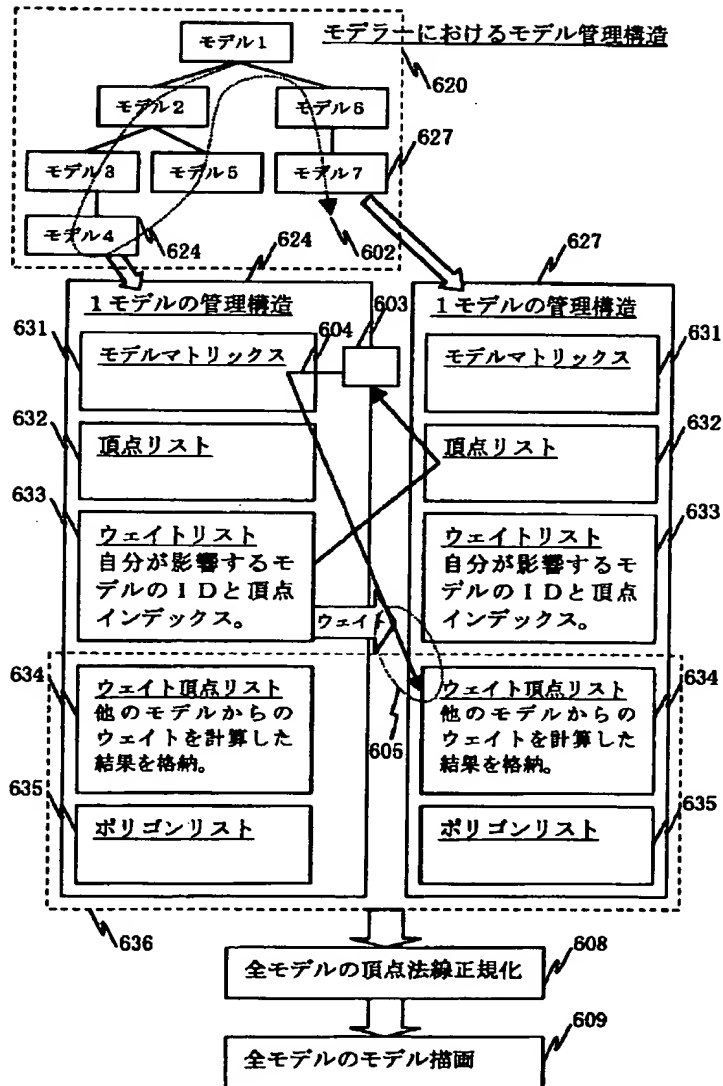
【図3】

従来例



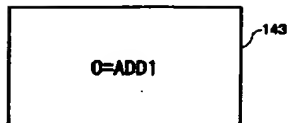
【図2】

従来例



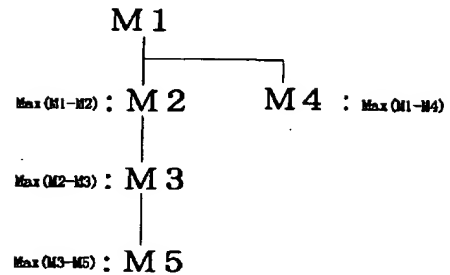
【図21】

未描画テーブルの例

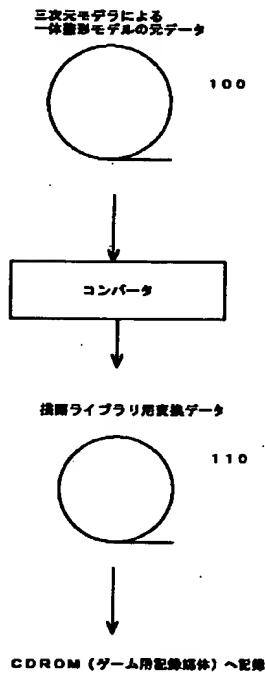


【図5】

一体型モデルのデータ構造例



【図10】



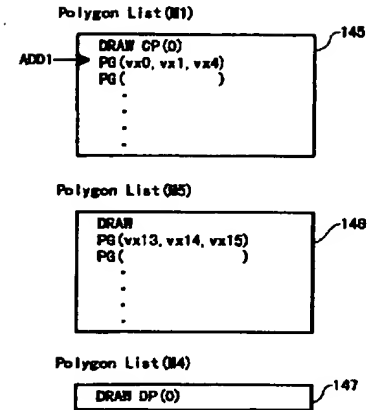
【図6】

各頂点のウェイトの例

モデル	影響 モデル	M1	M2	M3	M4	M5
	頂点					
M1	V101	100%				
	V102		30%	30%	40%	
	V103		30%	30%	40%	
	V104		40%	60%		
	V105		40%	60%		
	V106			100%		
	V107	100%				
	V108		60%	40%		
	V109		60%	40%		
	V110		40%	60%		
	V111		40%	60%		
	V112			100%		
	V113				100%	
M5	V501					100%
	V502					100%
	V503					100%
	V504					100%
	V505					100%
	V506					100%

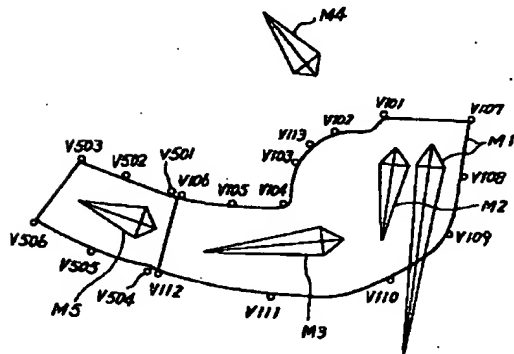
【図20】

ポリゴンリストの例



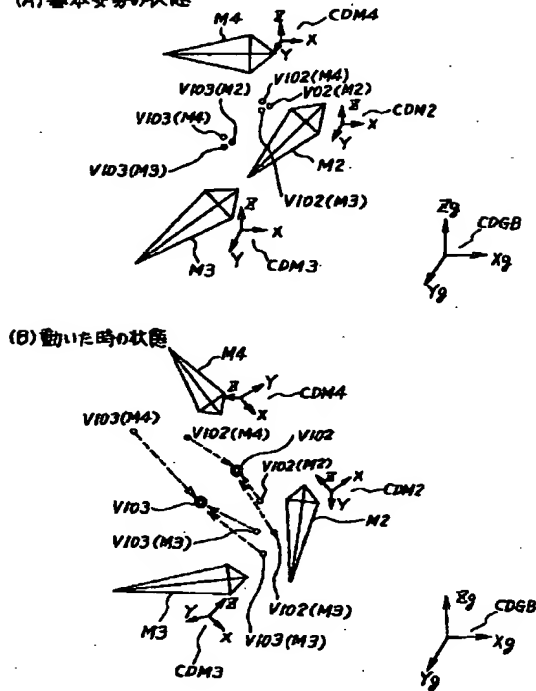
【図7】

一体整形モデル(エンベロープの移動例)



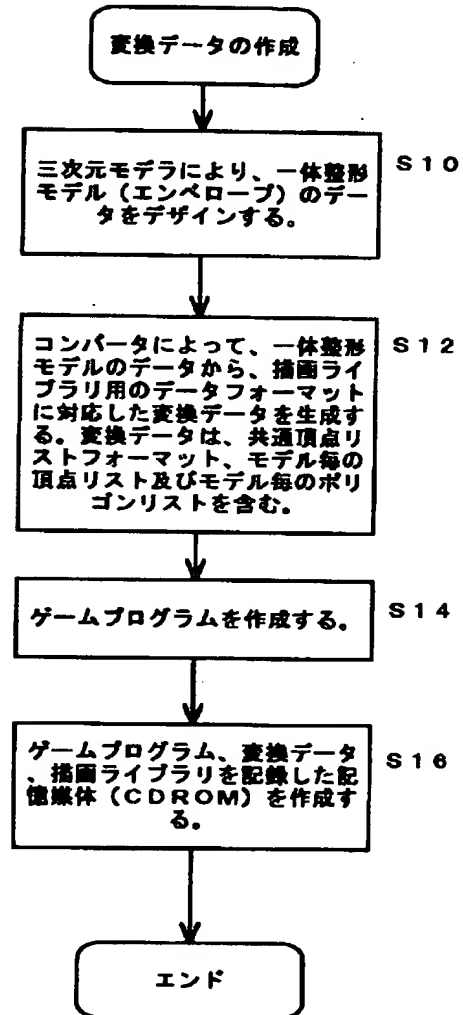
【図8】

ウェイト計算方法を説明する図
(A) 基本姿勢の状態



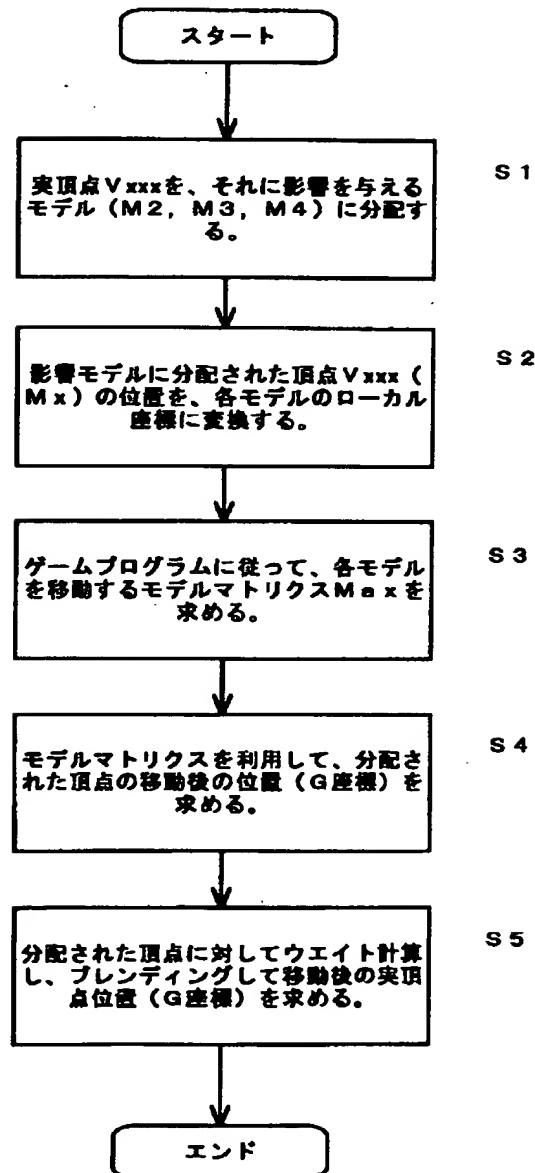
【図12】

コンバータ及び変換データ



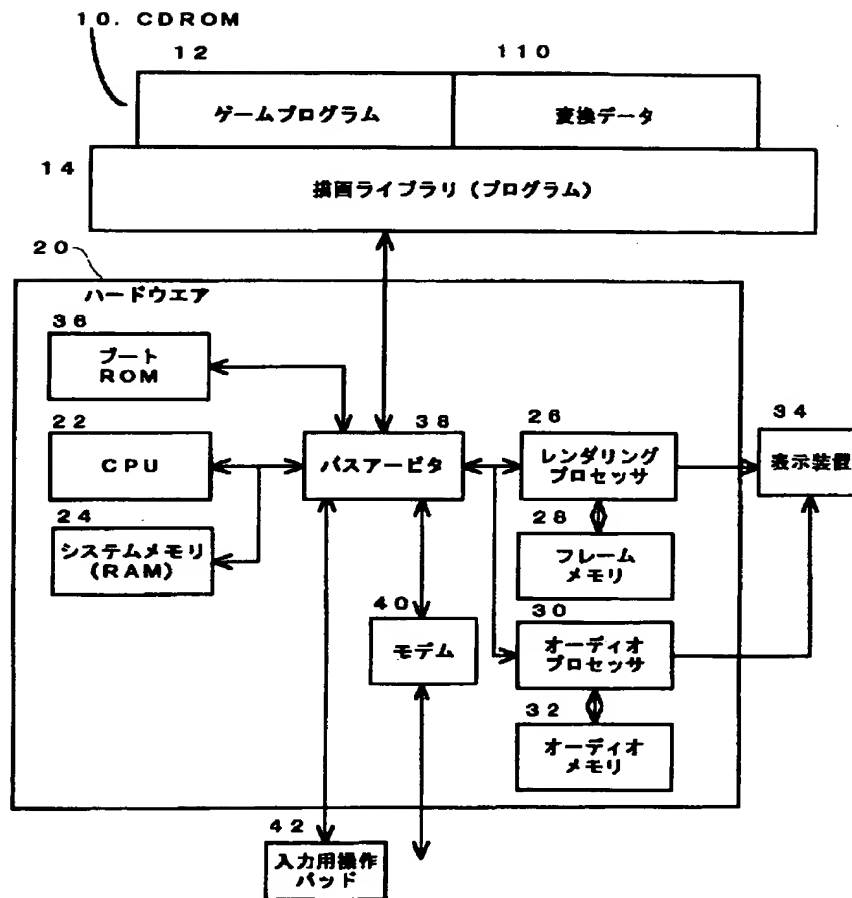
【図9】

移動計算とウェイト計算の手順



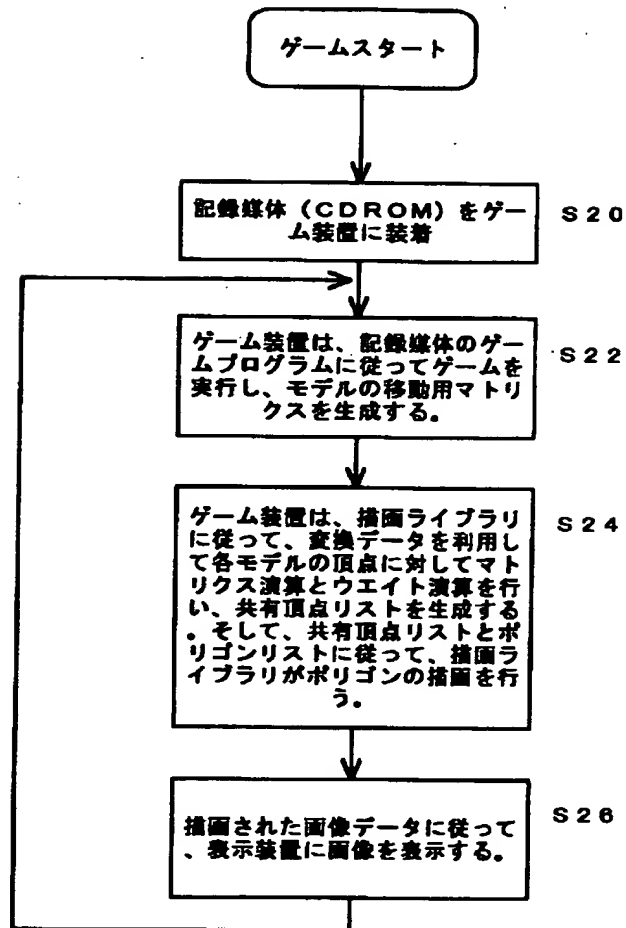
【図11】

記録媒体内の構成とハードウェアの関係



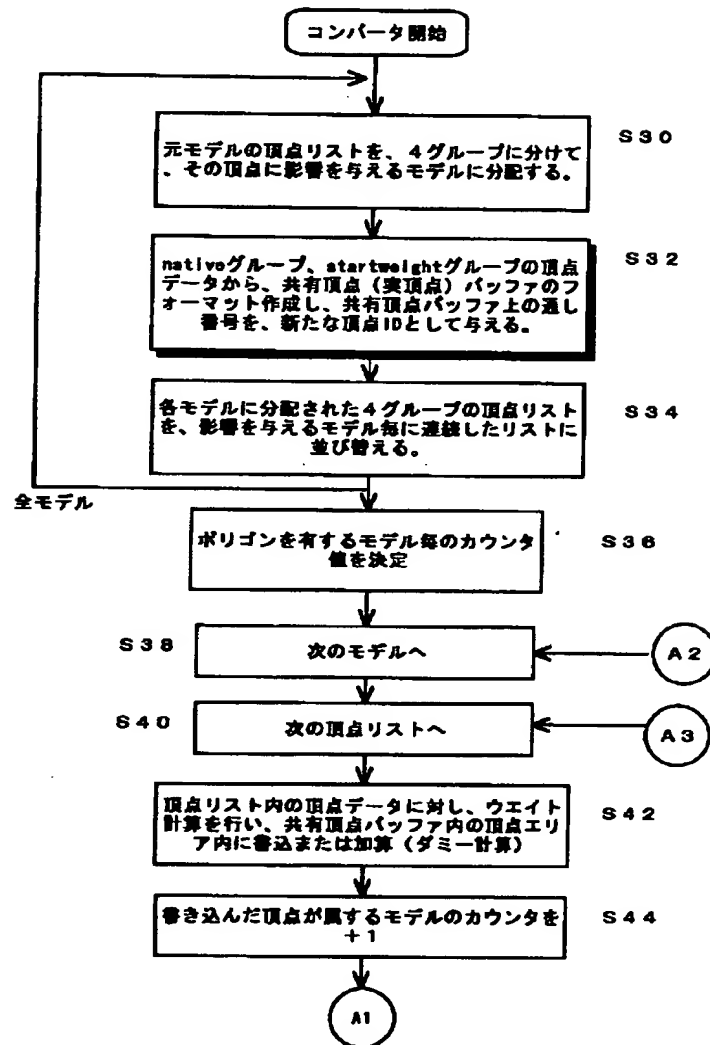
【図13】

ゲーム装置と描画ライブラリ



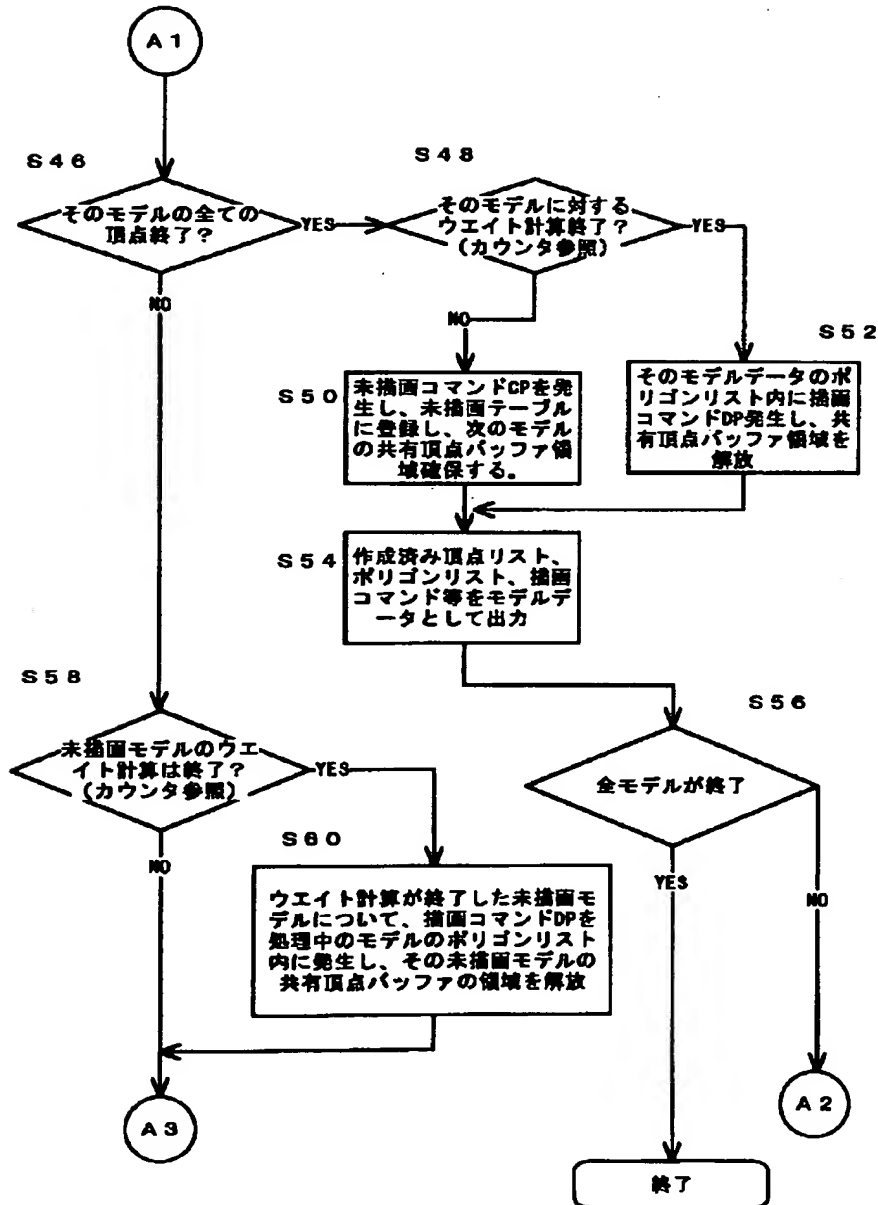
【図14】

コンバータの処理手順のフローチャート(1)



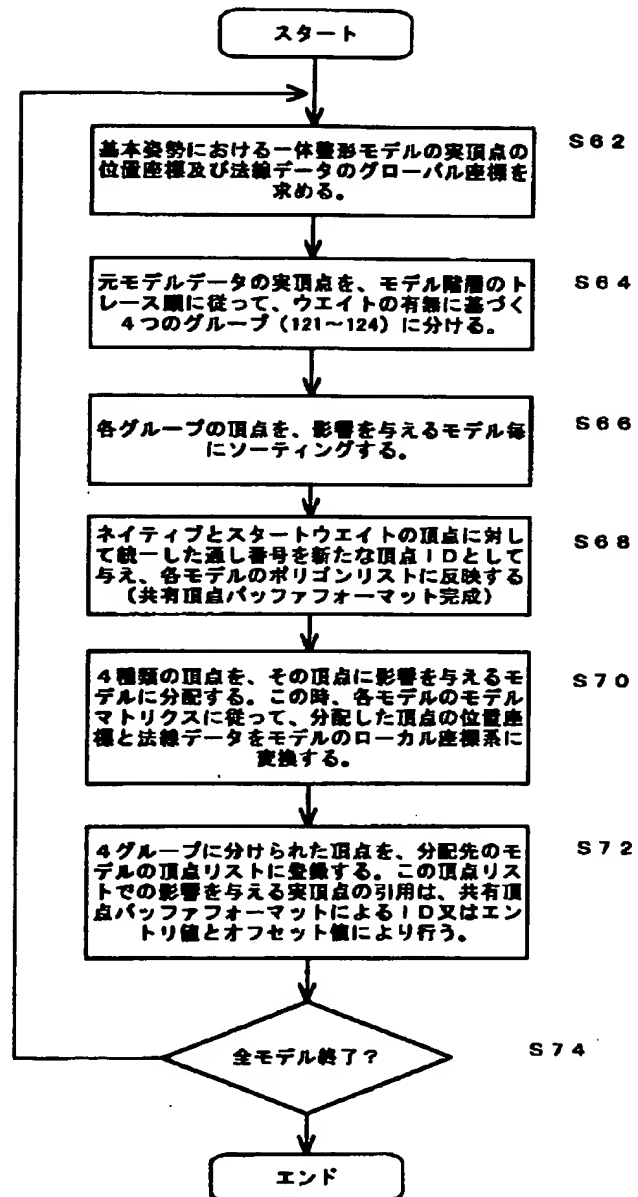
【図15】

コンバータの処理手順のフローチャート(2)



【図16】

頂点のモデルへの分配と頂点リストの生成



[illegible]

元頂点リスト → 4つのグループ → 共有頂点リスト

M1

V101 (x, y, z)
V102 (x, y, z)
V103
V104
V105
V106
V107
V108
V109
V110
V111
V112
V113

ネイティブ スタートケイト ミッドケイト エンドケイト

V101 (M1)

V102 (M2), V102 (M3), V102 (M4)
V103 (M2), V103 (M3), V103 (M4)
V104 (M2) V104 (M3)
V105 (M2) V105 (M3)
V106 (M3)
V107 (M1)

V108 (M2) V108 (M3)
V109 (M2) V109 (M3)
V110 (M2) V110 (M3)
V111 (M2) V111 (M3)
V112 (M2) V112 (M3)
V113 (M4)

M5

V501 (x, y, z)
V502
V503
V504
V505
V506

V501 (M5)
V502 (M5)
V503 (M5)
V504 (M5)
V505 (M5)
V506 (M5)

M1

NTV (ネイティブ)
M1 V101
V107
M3 V106
M4 V113
SW (スタートケイト)
M1 -
M2 V102
V103
V104
V105
V108
V109
V110
V111
V112
M3 -
M4 -

新たなID

vx0 ← ADD (M1)
vx1
vx2
vx3

vx4 ← ADD (M1) + 4
vx5
vx6
vx7
vx8
vx9
vx10
vx11
vx12

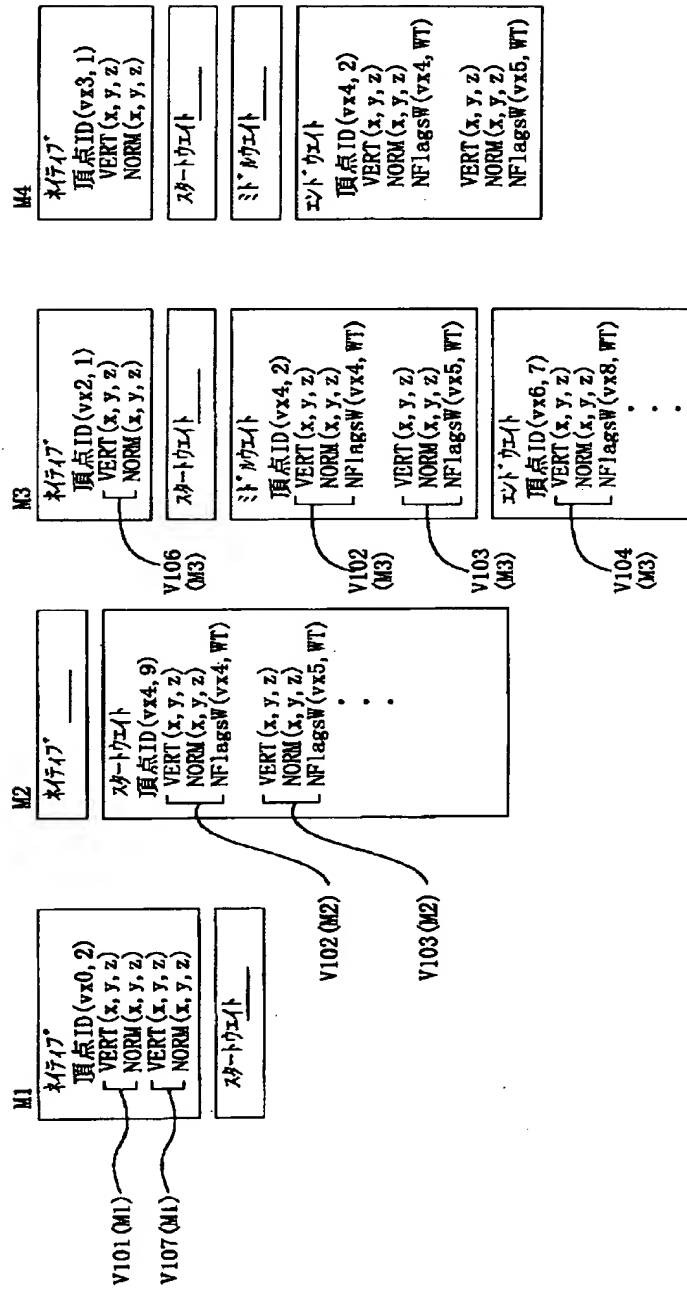
M5

NTV (ネイティブ)
M5 V501
V502
V503
V504
V505
V506

vx13 ← ADD (M5)
vx14
vx15
vx16
vx17
vx18 ← ADD (M5) + 5

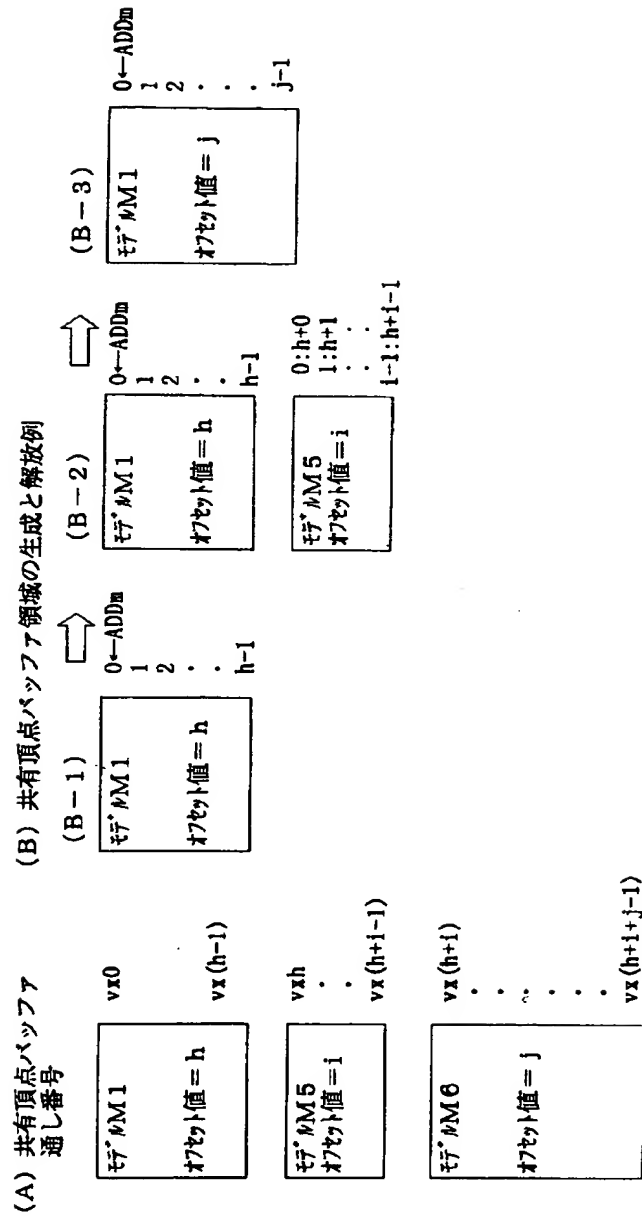
【図19】

頂点リストの例



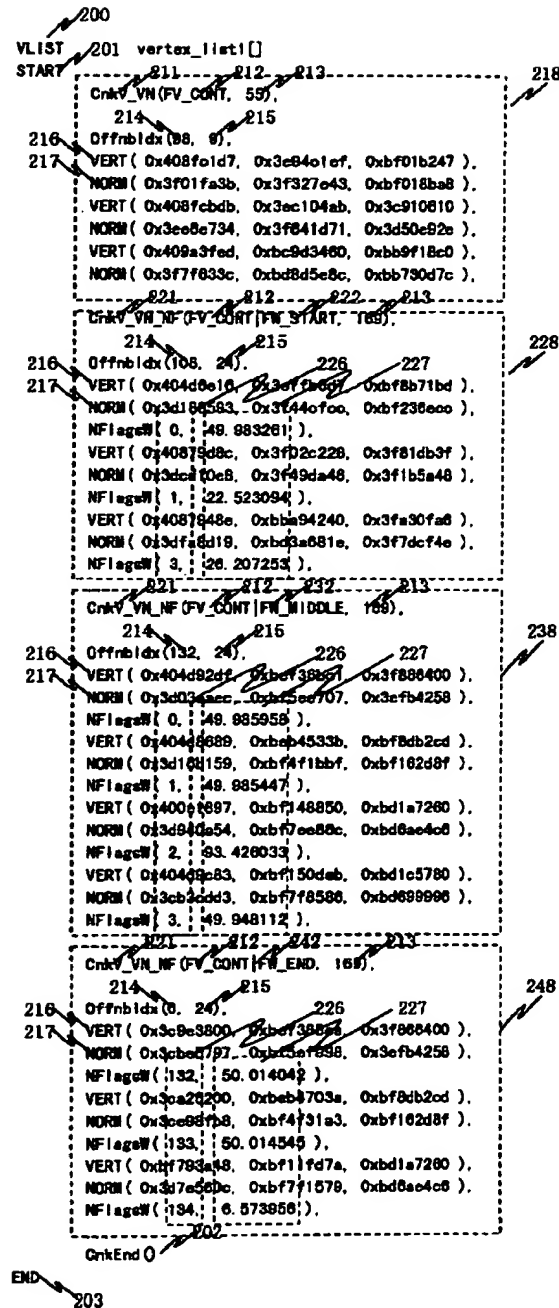
【図22】

頂点IDを説明する図



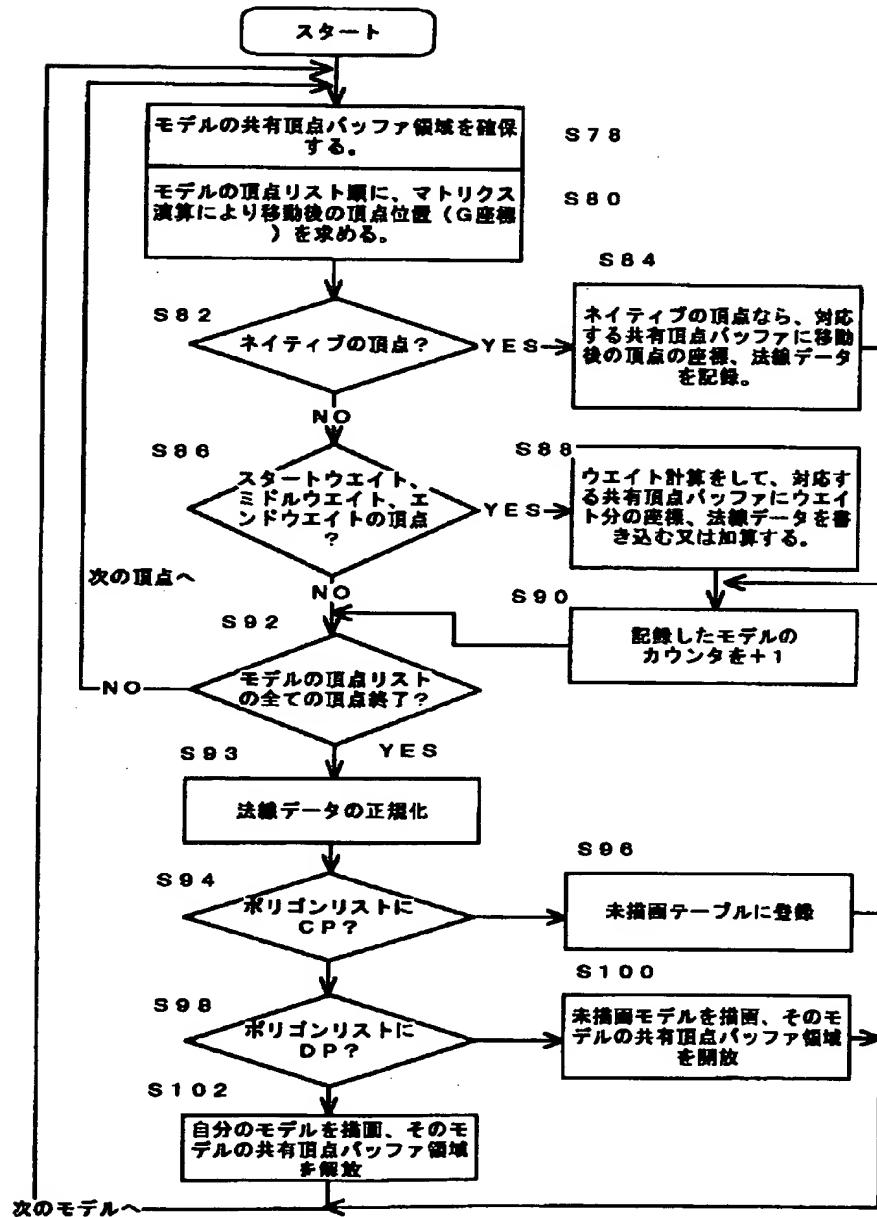
【図 23】

頂点リストの例



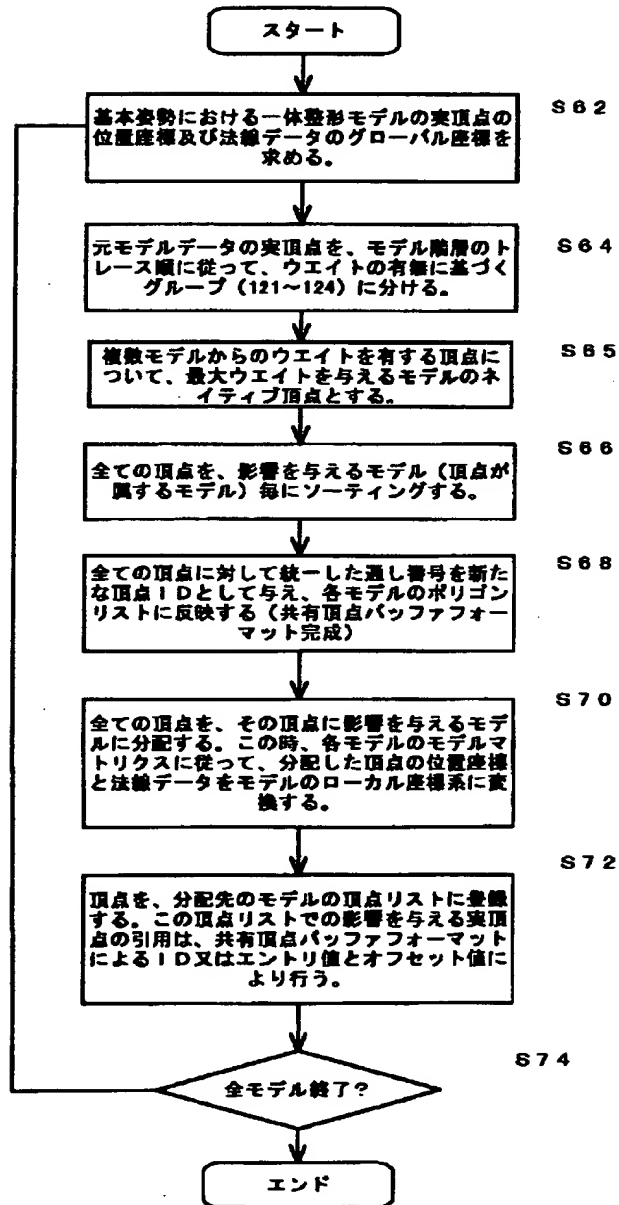
【図24】

描画ライブラリの手順のフローチャート図



【図25】

ワンウエイトの頂点のモデルへの分配と頂点リストの生成



フロントページの続き

Fターム(参考) 2C001 BC00 BC05 BC10 CB01 CB03
CC02
5B050 AA10 BA08 BA09 EA24 EA28
FA19
5B080 AA13 BA01
9A001 DD12 HH26 HH30 HH32 JJ76
KK45 KK62